Designing labor market recommender systems: the importance of job seeker preferences and competition*

Victor Alfonso Naya, Guillaume Bied, Philippe Caillou,
Bruno Crépon, Christophe Gaillac, Elia Pérennes, Michèle Sebag

May 31, 2021

Abstract

We examine the properties of a recommender algorithm currently under construction at the Public Employment Service (PES) in France, before its implementation in the field. The algorithm associates to each offer-job seeker pair a predicted "matching probability" using a very large set of covariates. We first compare this new AI algorithm with a matching tool mimicking the one currently used at the PES, based on a score measuring the "proximity" between the job seeker's profile or preference and the characteristics of the offer. We detail and discuss the trade-off between matching probability and preference score when switching from one system to the other. We also examine the issue of congestion. We show on the one hand that the AI algorithm tends to increase congestion and on the other hand that this strongly reduces its performance. We finally show that the use of optimal transport to derive recommendations from the matching probability matrix allows to mitigate this problem significantly. The main lesson at this stage is that an algorithm ignoring preferences and competition in the labor market would have very limited performances but that tweaking the algorithm to fit these dimensions improves substantially its properties, at least "in the lab".

*Contact: Alfonso Naya, LISN / INRIA: victor.alfonso-naya@inria.fr; Bied, CREST / UPSaclay / LISN / INRIA: guillaume.bied@ensae.fr; Caillou, UPSaclay / LISN / INRIA: caillou@lri.fr; Crépon: CREST, crepon@ensae.fr; Gaillac: CREST / TSE: christophe.gaillac@ensae.fr; Pérennes, CREST / Pôle emploi: elia.perennes@ensae.fr; Sebag, UPSaclay / LISN / INRIA / CNRS: sebag@lri.fr. This paper is the result of a collaboration with Pôle Emploi, the Public Employment service in France. We thank Camille Quéré, Chantal Vessereau, Cyril Nouveau, Sebastien Robidou and Thierry Foltier for their operational support. Christophe Gaillac acknowledges financial support from the grants ERC POEMH 337665 and ANR-17-EURE-0010. Authors retained full intellectual freedom throughout this process, all errors are our own.

1 Introduction

Job search platforms are becoming critical tools in the job market (Kässi and Lehdonvirta, 2018). Recommender systems, aiming at easing the matching between job seekers and offers, are often a key component of these platforms. These tools have advantages and drawbacks: they can bring important efficiency gains as well as potential threats in the labor market (Autor, 2001). In fact, these algorithms can jeopardize fairness as they can reproduce or even deepen inequalities of access to employment (see, *e.g.*, Li et al., 2020; Kasy and Abebe, 2021).

We are part of a long-term project with the French Public Employment Service (PES hereafter) which consists in building a recommendation algorithm, analyzing its properties and measuring its impact on the labor market. The goal is to improve the matching of job applications and vacancies thanks to a wealth of data. On the job seeker side, we have access not only to job search parameters, but also to detailed demographic data and past unemployment spells. On the vacancy side, available data includes the detailed parameters of each vacancy, the textual analysis of its content, as well as the hiring history of the firms that posted the offers. As shown in the paper, our algorithm performs well on standard metrics such as the recall@100: in 57% of cases our algorithm manages to place a future hire in the first 100 recommendations.

Leveraging the fact that we observe the sequence of applications as well as their outcomes, we associate the score resulting from the algorithm for each job seeker – vacancy pair, to a "matching probability". This matching probability individualizes the notion of a "suitable" vacancy pair (Belot et al., 2018). As described in the matching literature (see, *e.g.*, Chiappori and Salanié, 2016), there are unobserved characteristics on both sides of the pair which make a match uncertain. Later in the paper, we articulate this notion of matching probability with the competition between jobseekers on a vacancy.

We examine the properties of our recommendation algorithm "in the lab", *i.e.*, on a test sample of past data. This is an important phase of our analysis, which will be followed by the implementation of the recommender system in the field so as to measure its impact using a series of randomized control trials (RCT). Two questions are particularly important for us and guide our analysis. The first is the implicit losses in terms of work conditions that job seekers would suffer if they followed the recommendations from our algorithm. Indeed, the recommendations made by the algorithm are intended to maximize the chances of finding a job. However, job seekers have job search preferences such as reservation wage, job location, employment contract type and duration, etc. While our algorithm takes these job search criteria into account, it does not constrain the resulting recommendations to satisfy these preferences. On the contrary, the tools currently available in the PES to make referrals are based on these preferences: for each job seeker – vacancy pair, a preference score indicating the level of match between the job seeker's expectations and the offer's criteria is calculated. Vacancies are then ranked according to this score and the top ranked are referred to job seekers. Such filtering and ranking tools are quite common on online job search plateform (see, *e.g.*, Chilton et al., 2010). In this paper, we characterize each job seeker – vacancy pair as a lottery defined by its chances of success, which is the probability of matching post-processed from our algorithm and by a preference score inspired from the referral system currently used by the PES. Hence, we are able to compare two different approaches to recommender systems: the first one mimics the one from the PES and consists in maximizing the preference score, whereas the second one results from our own machine learning algorithm which amounts to maximizing the matching probability.

Our first result is that the offers from our recommendation algorithm have match probabilities which improve the return-to-employment chances. At the median, 12 sequential applications would be necessary to achieve a 50% hiring probability, whereas for the algorithm based on preferences, 54 would be needed. As expected, these gains in matching probability are obtained at the expense of a loss on the preference score. We document the nature of this loss and highlight its heterogeneity. While some job seekers would experience modest losses on the preference score, for others, potential losses are substantial. These losses correspond to several criteria of the preference score being unmet. In most cases, the wage criterion is unmet as well as the criterion for the type of contract sought. Less frequently, criteria related to the skill profile or to the primary target occupation are also unmet, leading to large losses on the preference score. The loss on the preference score is also heterogeneous among job seekers. Those who lose the most in terms of preference score are the more experienced job seekers, women, and those with low maximal geographical mobility. Symmetrically, there is also a strong heterogeneity in matching probability gains. Job seekers who gain the least in terms of matching probability are those with higher reservation wage and who have longer unemployment duration.

We also examine recommendations that would be generated by maximizing not the matching probability or preference score, but their product. This can be interpreted as the maximization of an expected preference score, weighting the preferences for vacancies by their matching probabilities. Using such an objective would be more in line with the literature on directed search (see *e.g.*, Wright et al., 2017; Belot et al., 2018). We show that compared to the standard use of our recommendation algorithm, the matching probabilities in this case are only slightly lower. But, on the other hand, the losses in preference score are much lower. Although it is dependent on the specific case considered, this result is encouraging as it shows that it is possible to exploit the matching potential of the recommender system in a way which preserves individual preferences. These "in the lab" results highlight the importance of aligning the scores on which recommendations are made with the expectations of job seekers.

The second issue we address is congestion. We interpret the matching probability as an individual and instantaneous probability, independent of the competition in the labor market. As our algorithm generates recommendations for each job seeker, it is likely that the same vacancy is recommended several times. Hence, for some vacancies, recommendations can increase competition for the job. Reversely, they could decrease the number of candidates applying to other vacancies. We document the distribution of the number of referrals received by each vacancy and illustrate it by their Lorenz curve (see also Li et al., 2019; Chen et al., 2019, for related approaches). This evidence illustrates how important it is to take into account this competition between job seekers. We define a competition-adjusted matching probability which involves not only the initial individual matching probability but also the distribution of matching probabilities of job seekers to whom the vacancy is also recommended. We show that this competition-adjusted matching probability is substantially lower than the instantaneous matching probability. We conclude that ignoring competition in the labor market would substantially reduce the performance of the algorithm.

To solve this problem, we modify the way we derive sets of recommendations from the matching probabilities matrix. Instead of taking for each individual the vacancy that maximizes the matching probability, we identify it using optimal transport (see, *e.g.* Dupuy and Galichon, 2014; Li et al., 2019, for the use of optimal transport in other matching contexts, especially the matrimonial market). That is, we consider a global objective which imposes to distribute the job seekers equally on all the vacancies and vice versa, instead of a succession of individual objectives. We detail the changes associated with the switch from the individual to the global objective. We document the resulting variations both on the individual matching probability and on the competition-adjusted matching probability. Making recommendations based on optimal transport rather than on individual objectives leads to apparent losses in matching probabilities. But it also leads to a better distribution of job seekers over vacancies. As a result, the adjustment of matching probabilities to account for competition in the labor market is less important when optimal transport is used. We show that the distribution of competition-adjusted matching probabilities resulting from optimal transport has first order stochastic dominance over the one resulting from individual objectives (*i.e.*, without using optimal transport).

This paper is related to several strands of the literature. The first is concerned with the use of recommendation algorithms in the labor market (see, *e.g.*, Horton, 2017; Li et al., 2020). Our contribution is to build and study the properties of a highly performing state of the art machine learning algorithm relying on very large amounts of data. One of the key insights of our analysis is the following: recommendation algorithms focus on a precise objective - most of the time, improving the chances of a match - which might be different from the jobseekers'. This disconnection between the two objectives can result in substantial losses. Indeed, it is possible that, following the recommendations made to them, some individuals focus their search on vacancies that are far from their preferences. The second one pertains to job search behavior in relation to directed search (see, *e.g.*, Belot et al., 2018, 2019; Wright et al., 2017). Our modeling of vacancies as a lottery with a matching probability and a preference score is directly inspired by the literature on directed search. We identify the components of these lotteries and document their heterogeneity. Our work is also related to the literature on preferences for various job attributes and how ignoring its heterogeneity can lead to frictions (see, *e.g.*, Banerjee and Chiplunkar, 2019; Mas and Pallais, 2017). The third studies the congestion of the labor market and the importance of competition (see, *e.g.*, Lazear et al., 2018; Gee, 2019). We show that the congestion of applications on vacancies is high and that our algorithm increases it. We also show that once congestion is taken into account, the measured performance of the algorithm is much lower. The performance of the algorithm, applied at scale, would thus be substantially lower than if it were applied on a reduced population. Finally, our work is related to the literature on optimal transport (see, *e.g.*, Galichon, 2016). We show that using optimal transport to obtain recommendations from our matching probability matrix solves the previous congestion problem to a large extent and restores the properties of the algorithm.

The paper is organized as follows. Section 2 presents the preference-based and prediction-based recommender systems. Section 3 compares the different recommender systems using the preference score and the matching probabilities. Section 4 then analyses the congestion in standard recommendations and describes our solution based on optimal transport. Appendix A provides supplemental figures and tables while Appendix B gives additional details on the standard recommendation algorithm.

2 Two recommender systems

2.1 A preference-based recommender system

The French PES has developed a matching algorithm which is used to 1) suggest relevant vacancies to job seekers and 2) suggest relevant job seekers to recruiters.

This matching algorithm is used for three purposes. First, it is used by caseworkers to recommend job ads to job seekers. Second, it is used by caseworkers to recommend candidate profiles to recruiters. Third, it is also integrated into each job seeker's and recruiter's personal space on the PES website. The matching is done on few criteria available both on the jobseeker side and the recruiter side (*e.g.*, salary desired by the jobseeker vs. salary offered in the job ad). Table 1 lists the characteristics used to define these criteria.

For the purpose of this study we built a recommender-system inspired by the one used at the PES and based on the same list of criteria. But as we only had partial access to the recommender system used by the PES, it is likely to be different.

In practice, the selected criteria are matched with the same exact criteria on the recruiters' side (*i.e.*, profile required in the vacancy and characteristics of the proposed job). For each characteristic k, a consistency criterion $c_k(i, j) \in [0, 1]$ is defined, corresponding to whether characteristic k of the jobseeker profile is consistent with the characteristic in the vacancy. For example, for "reservation wage", the criterion takes the value 1 if the wage offered in the vacancy is above the reservation wage in the jobseeker's profile. For "geographic mobility", the criterion takes the value 1 if the distance between the job location and the jobseeker home-place is below the jobseeker's maximum commuting distance. The exact criteria used at the PES share the same principles but allow for more smoothness in the definition of the sub-criteria used for each characteristic.

"Skills" and "Occupation" are important criteria. In his/her individual profile, a jobseeker enters a primary occupation she is searching for, as well as a set of skills. The criterion "Occupation" takes value 1 if the occupation the jobseeker seeks is the exact same occupation as the one entered in the vacancy, or if it is a close occupation. This proximity is defined according to an expert-based matrix made available by the PES. The criterion "skills" corresponds to the adequacy between the skills listed in an individual profile and the skills listed in the vacancy.

From the jobseeker's point of view, there are two kinds of inputs to the algorithm, which are gathered in column (1) and (2) of Table 1. Criteria in column (1) are listed as "Profile". The underlying characteristics are considered difficult to adjust in the short-term. Many of these criteria can be seen as components of the utility function of jobseekers. This is especially true for the criteria in column (2) about job search characteristics.

Profile (CV)	Given weight	**Search criteria**	Given weight
Skills	1000	Occupation	1000
Diploma	100	Working hours	100
Languages	100	Reservation wage	200
Driving license	100	Geographic mobility	300
Years of experience	100	Duration and type of contract	10

Table 1

Each criterion is then associated with a weight and the final matching score is the weighted sum of each single match between the criteria of applicants and the criteria of recruiters:

$$\Pi_w(i,j) = \sum_{k=1}^{K} w_k c_k(i,j) \Big/ \sum_{k=1}^{K} w_k, \tag{1}$$

where w_k is a set of weights, which is the same for the whole population.

The set of weights we consider is again inspired by the expert-defined one chosen by the PES. It gives large weights to skills and occupation, reflecting the priority given to these criteria in the PES algorithm.

Notice that we can also define two sub-scores corresponding to the decomposition of criteria in the "profile" criteria and "search" criteria. This is given by:

$$\Pi_{w,\text{Profile}}(i,j) = \sum_{k \in \text{Profile}} w_k c_k(i,j) \Big/ \sum_{k \in \text{Profile}} w_k \tag{2}$$

for the profile characteristics: skills, diploma, languages, driving license and year of experience, and by

$$\Pi_{w,\text{Search}}(i,j) = \sum_{k \in \text{Search}} w_k c_k(i,j) \Big/ \sum_{k \in \text{Search}} w_k \tag{3}$$

for the job search characteristics, namely: occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract.

2.2 A prediction-based recommender system

2.2.1 Learning to rank hires over other jobseeker-offer pairs

The present section describes the outline of a prediction-based recommender system implemented and optimized for the problem at hand. It focuses on the recommendation of offers to jobseekers; the reverse can be done according to similar principles.

In a nutshell, machine learning techniques aim at leveraging the experience of past jobseekers and recruiters, as recorded by the PES, to design effective recommendation strategies. We observe a large set of jobseekers i and job ads j in the French Rhône-Alpes region from ISO weeks 1 to 48 of 2019. The number of unique jobseeker search sessions (resp. job ads) is 1,181,902 (resp. 516,776); on average, 610,986 jobseekers (resp. 129,642 job ads) are active a given week.

The characteristics of jobseeker i will be denoted by X_i, and those of offer j by Y_j. The variety of features on both sides is described in Appendix table 1. Aside from languages, they cover the inputs to the preference-based system described above with finer granularity whenever relevant, as well as a variety of other features. The vector X_i includes search criteria, geographic location, experience, skills, unemployment duration, applications in the last six months, status *w.r.t.* the minimum allowance, various individual and post-code level socio-demographic features. On the job vacancy side, Y_j contains occupation at several levels of granularity, offered wage, required experience, contract type, workplace location, required hard and soft skills (as reduced by singular value decomposition as well as raw inputs), textual descriptions of the job ad and firm (reduced by singular value decomposition), establishment size, number of applications to the job ad and to the establishment in the last six months, time since the offer was posted, etc.

Crucially, we observe whether or not a jobseeker i was hired on job ad j. We observe 75,744 successful matches in the data. Observations from week 1 to 43 of 2019 are used as a training set (representing 66,914 matches); while weeks 44 to 48 (representing 8,830 matches) are used as a test set to evaluate the quality of recommendations.

The metric to be optimized (according to which models are evaluated and their hyperparameters selected) is the so-called recall@k on the test set: the proportion of jobseekers i in the test sample who where hired on a vacancy j in the top k recommendations (among offers available the week of the match).

While the recall@k defines a performance metric to evaluate algorithms on the test set, it is intractable for direct optimization. Drawing on the learning to rank literature, we propose to learn a dissimilarity score $S_{i,j}$ between jobseeker i and job ad j as a function of the jobseeker's and the ad's characteristics. Given a jobseeker i and two offers j and j', we wish S_{ij} to be lower than $S_{ij'}$ if i matched with j and not with j'. This motivates the minimization on the training set of the so-called triplet margin loss, corresponding to the following objective:

$$\min_S \sum_i \sum_{j' \neq j^*(i)} [S_{i,j^*(i)} - S_{i,j'} + \eta]_+$$

where $\eta > 0$ is a scalar hyperparameter, $[x]_+ = \max(x, 0)$, the outer sum ranges over all jobseekers with matches, the inner one over all job ads, and $j^*(i)$ is the job ad with which i actually matched. This expression aims at separating, for all jobseekers, the scores associated to ads they matched with from the scores associated to all other job ads by a margin of at least η.

Given jobseeker characteristics x_i and job ad characteristics y_j, the score S_{ij} is parametrized as:

$$S_{ij}(x_{i,j}) = \phi(x_i)^T A \psi(y_j),$$

where $x_{i,j} = (x_i, y_j)$, ϕ, ψ are feed-forward neural networks with several layers, and A is an affinity matrix. Feed-forward neural networks are flexible, differentiable functions commonly used in the machine learning literature, that handle high-dimensional features well when given large datasets. Their definition is briefly recalled in the appendix (section B.1); the interested reader may consult Goodfellow et al. (2016) for a textbook treatment.

In this context, $\phi(x_i)$ and $\psi(y_j)$ may be understood as latent variables describing i and j. A can be interpreted as an affinity matrix: the parameter $A_{m,n}$ represents the complementarity between dimension m of the jobseekers' latent space and dimension n of the ad's latent variables. The latent space is of size 872 for both jobseekers and offers, although ϕ, ψ and A are given a block-wise structure which incorporates domain knowledge (three blocks corresponding to geography, to skills, to other factors) and reduces the number of parameters. In other words,

$$S_{ij} = \phi_g(x_{ig})^T A_g \psi_g(y_{jg}) + \phi_s(x_{is})^T A_s \psi_s(y_{js}) + \phi_o(x_{io})^T A_o \psi_o(y_{jo}),$$

where subscripts g, s, and o correspond to "geography", "skills" and "other".

The parameters that are optimized during training are the parameters of the transformations from the observed to the latent variables (*i.e.* the weights of the neural networks) and the affinity matrices. As is customary in the machine learning literature, the minimization of this (non-convex) objective function is done by mini-batch stochastic gradient descent. For computational efficiency, pairs that are not matches are very aggressively uniformly subsampled. The exact specification of the networks and of the training procedure are described in Appendix B.

2.2.2 Calibration of a matching probability using the proximity score

As detailed in Section 2.2.1, based on observables $X_{i,j}$ for each jobseeker (user) / job offer (item) pair (i, j), our recommender system produces a score

$S_{i,j} := S(X_{i,j})$. To a jobseeker i, top offers are recommended by sorting offers according to scores $S_{i,\cdot}$.

However, one might want to characterize the information contained in $S_{i,j}$ about the true probability of i being matched with j, denoted by $\mathbb{P}(M_{i,j} = 1|X_{i,j} = x_{i,j})$. The true conditional probability is an object which is difficult to estimate while taking into account the high dimensionality of the problem. In this section, we provide a framework which allows to calibrate the ML score produced by the recommender system and learned on past matches to learn about the true matching probability.

Our score is obtained from a training sample and aims at ranking the pairs of jobseekers and offers according to their chance of leading to a match. We want to give an interpretation to this ranking in terms of hiring probability. More precisely, we want to identify the probability of a match conditional on the score $S_{i,j}$ for a randomly chosen pair (i, j) that would be put "artificially" in contact. Let us denote this latent match variable by $M^*_{i,j}$. We observe hires $M_{i,j} \in \{0, 1\}$ conditional on contact $C_{i,j} = 1$. We have access to a file recording all the pairs of job seekers and offers that have been put in contact by the PES, at the initiative of the job seeker, the recruiter, or the caseworker: the so-called "MER" file.[1]. For all these pairs, we also observe the dates of the contact as well as whether there was ultimately a match. We want to identify $P(M^*_{i,j} = 1|S_{i,j}, C_{i,j} = 1)$, but can only identify $P(M_{i,j} = 1|S_{i,j}, C_{i,j} = 1)$. We face a classical selection problem. To deal with this issue we make two assumptions. Let's call $X_{i,j}$ the set of variables at our disposal from which we build our score. Our assumptions are the following:

- **Assumption 1**. Conditional independence assumption:

$$M^*_{i,j} \perp C_{i,j} \mid X_{i,j}.$$

  Given how large the set of covariates we are starting from is, this assumption makes sense.

- **Assumption 2**. $S_{i,j}$ (which is only a function of $X_{i,j}$) is a sufficient statistic: $P(M^*_{ij,} = 1|S_{i,j}, X_{i,j}) = P(M^*_{ij,} = 1|S_{i,j})$

It is straightforward that under these two assumptions $P(M_{i,j,} = 1|S_{i,j}, C_{i,j} = 1) = P(M^*_{ij,} = 1|S_{i,j})$. Thus, we now describe the procedure we follow to identify $P(M_{i,j} = 1|S_{i,j}, C_{i,j} = 1)$.

We assume that we observe a sample of *i.i.d.* observations of job seekers' characteristics and, for each jobseeker i_0, sequential job searches on vacancies identified by $C_{i_0,j} = 1$. We denote this set of vacancies by $\mathcal{J}(i_0) = \{j|C_{i_0,j} = 1\}$

[1]MER is for the French "mise en relation" which means being "put in contact".

and rank them by order of application $1(i_0), 2(i_0), \ldots, j^{max}(i_0)$. We denote by $N(i_0)$ the number of vacancies in $\mathcal{J}(i_0)$ (which is also the rank of the last application in the set). The sequence of applications ends either with a match on the last offer or with exogenous censorship: $M_{i_0,j^{max}(i_0)} \in \{0,1\}$ and $M_{i_0,j} = 0$ otherwise.

This sequential search model is thus akin to a sequence of binary choice models. Thus, we consider a discrete duration model formulation, where we model the hazard rate as a logistic transformation of the scores $S_{i,j}$,

$$\mathbb{P}\left(M_{i,j(i)} = 1 | M_{i,1(i)} = 0, \ldots, M_{i,j-1(i)} = 0, \mathcal{J}(i), S\right) = \Lambda(\alpha_j + \beta S_{i,j(i)}),$$

where $S := (S_{i,1(i)}, \ldots, S_{i,j^{\max}(i)})$ denotes the sequence of scores of the offers and Λ is the logistic transformation. The dependence of (α_j, β_j) in the state j takes into account weariness effects. Note that, in this model the probability that a jobseeker matches on the n-th offer of the sequence S is given by

$$\begin{aligned}&\mathbb{P}(M_{i,1(i)} = 0, \ldots, M_{i,n-1(i)} = 0, M_{i,n(i)} = 1 | \mathcal{J}(i), S)\\ &= \Lambda(\alpha_n + \beta S_{i,n(i)}) \prod_{j=1}^{n-1} (1 - \Lambda(\alpha_j + \beta S_{i,j(i)}))\end{aligned}$$

Taking into account completed and censored spells, the log-likelihood function, conditional on the scores produced by the recommender system (see, *e.g.*, Jenkins et al., 1995), is given by

$$\begin{aligned}\mathcal{L}(\alpha, \beta | C, M, S) = &\sum_{i=1}^{n} \sum_{j=1}^{n} C_{i,j} (M_{i,j} \ln(\Lambda(\alpha_{r(i,j)} + \beta S_{i,j})\\ &+ \sum_{i=1}^{n} \sum_{j=1}^{n} (1 - M_{i,j}) \ln(1 - \Lambda(\alpha_{r(i,j)} + \beta S_{i,j}))),\end{aligned}$$

where $r(i,j)$ is the rank of vacancy j in $\mathcal{J}(i)$.

Note that, if we omit the rank of vacancy j, this expression is symmetric in i and j. Thus, we could see as well this expression as the result of a process in which firm posting vacancy j sequentially considers candidates applying to the vacancy. This simple remark shows that there is a simple generalisation of the former expression to account for $r(i,j)$ the rank of offer j in the application set of jobseeker i, but also $q(i,j)$ the rank of i in the applicant pool for vacancy j. The expression of the likelihood in this case writes as

$$\begin{aligned}\mathcal{L}(\alpha, \beta | C, M, S) = &\sum_{C_{i,j}=1} M_{i,j} \ln(\Lambda(\alpha^{v}_{q(i,j)} + \alpha^{js}_{r(i,j)} + \beta S_{i,j})\\ &+ \sum_{C_{i,j}=1} (1 - M_{i,j}) \ln(1 - \Lambda(\alpha^{v}_{q(i,j)} + \alpha^{js}_{r(i,j)} + \beta S_{i,j})),\end{aligned}$$

where α^v and α^{js} are the sequences of "weariness" effects for vacancies and jobseekers.

Table 2 presents the results of the estimation of the parameter α and β. Each column corresponds to a different specification for the function Λ and choices regarding α. Column (1) considers a simple specification in which we just enter the score. As expected, the estimated coefficient is positive and significant. Column (2) adds effects corresponding to the ranks of vacancy j in the application set of jobseeker i. While the coefficient is somewhat smaller, we observe weariness effects: the probability of a match decreases as the number of application increases. The last column of the table also adds effects corresponding to the number of candidates met by the firm. In this column (3), while the coefficients corresponding to $S_{i,j}$ is larger, we observe again that the chances of a match decreases as the number of applicants the firm met increases.

Table 2: Estimates of the calibration model parameters

Method	(1)	(2)	(3)
Intercept	-4.179***	-3.591***	-3.297***
$S_{i,j}$	0.061***	0.049***	0.08***
Nb. appli. 2-5		-0.688***	-0.678***
Nb. appli. 6-10		-0.908***	-0.888***
Nb. appli. 11+		-1.204***	-1.19***
2-5 interviews			-0.499***
6-10 interviews			-0.8***
More than 11 inter.			-0.95***

Notes: 42,819 observations without restrictions on the sector of activity. Significance levels: 1% : ***. $\Pi_{i,j}$ is the preference score, $S_{i,j}$ the matching score, "Nb. appli." are dummies for the ranking of the application j in the list of applications of jobseeker i. "Second interview" and "More than second inter." are dummies for the ranking of the candidate j in the list of recorded interviews for job offer j.

From now on, we define an individual matching probability $p(i, j)$ based on the score and estimates in column (1) of table 2. We refer to $p(i, j)$ as the "matching probability score":

$$p(i, j) = \Lambda(\alpha + \beta S_{i,j}) \tag{4}$$

Table 3 presents some elements of the distribution of the related probabilities using estimates of α and β in column (2) of Table 2. For each jobseeker i, we consider in each column a specific type of vacancies $j(i)$ and report quartiles of the distribution of the implied probability $p_{i,j(i)}$ given by (4). Column (1) is the most important and considers the best vacancy to recommend to each jobseeker, i.e., the vacancy $j(i)$ from the full set of available vacancies maximizing $p(i, j)$. As it can be seen from the table, the minimum probability is 0.04 and the maximum 0.11. While this might not be perceived as not so large and so different, it corresponds to substantial differences in the chances of finding a job. The number of applications needed to be hired with a given probability is a good illustration. A probability of 0.11 implies that 6 applications would be needed to be hired with a probability of 0.5, and 12 and 20 to be hired with probabilities of 75% and 90% respectively. These numbers of applications are respectively 17, 34 and 56 when the probability of finding a job is 0.04.

Columns (2) and (3) consider as $j(i)$ respectively the vacancy on which the job seeker was hired and the best vacancy (with respect to $p(i, j)$) with which the job seekers was put in contact, as recorded in the PES information system. While obviously the distribution in column (1) dominates the distributions in column (2) and (3), these two distributions are close and look also similar to the distribution in column (1). This is an interesting point. This tends to show that, except at the very bottom of the distribution, for most jobseekers there are no matches that would have been substantially more likely than the one they have made. This means that for most jobseekers there are no obvious gains in terms of increased chance of returning to employment that they would miss. Column (4) of the table shows the difference between the best vacancy and the vacancy on which the applicants were hired. Although the maximum value is quite high (0.06), for 75% of the sample it is less than 2%. The distribution of the differences between the recommended vacancy and the worst vacancy is shown in the last column of the table. It shows larger differences, but still measured. The maximum value is 0.07, but for 50% of the sample it is less than 3%. This reinforces the feeling that the gain of the algorithm is to quickly identify relevant vacancies that could lead to a match. The effect of the algorithm is to reduce friction in the labor market and to lead to a reduction in unemployment spells. For the majority of individuals, the use of the algorithm should result in a faster return to employment, but the impact is likely to quickly vanish.

2.3 Recommendation based on a score

The derivation of a set of recommendations from a matching score $\chi_{i,j}$ is straightforward. To make k recommendations to jobseeker i_0 a simple and intuitive solution is to pick the k vacancies with the largest score $\chi_{i_0,j}$. We note $\mathcal{J}_k^*(\chi, i_0)$, or $\mathcal{J}_k^*(i)$ if there is no ambiguity, the set of vacancies recommended to jobseeker i_0 when using the score χ.

As mentioned in Section 2.2, to measure the performance of the recom-

Table 3: Distribution of predicted probabilities for different types of vacancies

	$P(S^*)$	$P(MER^+)$	$P(MER^*)$	$P(S^*) - P(MER^+)$	$P(S^*) - P(MER^-)$
Min	0.04	0.01	0.01	0.00	0.00
Q25	0.05	0.04	0.02	0.01	0.02
Q50	0.06	0.04	0.03	0.01	0.02
Q75	0.06	0.05	0.04	0.02	0.03
Max	0.11	0.09	0.09	0.06	0.07

Notes: $P(S^*)$ is the matching probability of the offer which has the highest calibrated matching probability among the set of all potential offers. $P(MER^+)$ is the matching probability of the offer which has the highest calibrated matching probability among the set of applications where the jobseekers were hired. $P(MER^*)$ is the matching probability of the best vacancy (with respect to $p(i,j)$) with which the jobseekers were put in contact. $P(MER^-)$ is the lowest matching probability among the vacancies with which the jobseekers were put in contact.

mender system χ, we define an individual success variable for each jobseeker

$$M_i^k(\chi) = \mathbb{1}\left(\sum_{j \in \mathcal{J}_k^*(\chi,i)} M_{i,j} = 1\right), \tag{5}$$

where $\mathbb{1}(\cdot)$ denotes the the indicator function. The global performance of such a recommender system is assessed using the "recall@k", which is the proportion of jobseeker who were hired on one of the top-k recommendations:

$$\text{recall@}k(\chi) = \frac{1}{N}\sum_i M_i^k(\chi). \tag{6}$$

We apply this framework to our two scores: the preference score and the probability score. Thus, for each jobseeker we define two sets of recommendations: a set based on the preference score $\Pi_w(i,j)$, mimicking the score used at the PES and our probability score based on our ML estimation.

Figure 1 shows the performance of our recommender system. The figure on the right panel compares the performances in terms of recall@100 on the test set of different recommender systems for the region Rhône-Alpes. Progressively including additional variables (such as previously considered vacancies) yields huge improvements on the recall. Several recommender systems are considered. The first one ("fixed weights") corresponds to the preference-based recommender system inspired from the PES' current one. It uses the adequacy score defined by (1) to rank the set of available offers for each jobseeker. For the recall@100, we select the first 100 offers according to this criterion and determine the proportion of jobseekers in the test set hired on one of their 100 best offers. The recall@20 corresponds to the same quantity but considering only the 20 best offers. As the graph shows, this proportion, of the order of

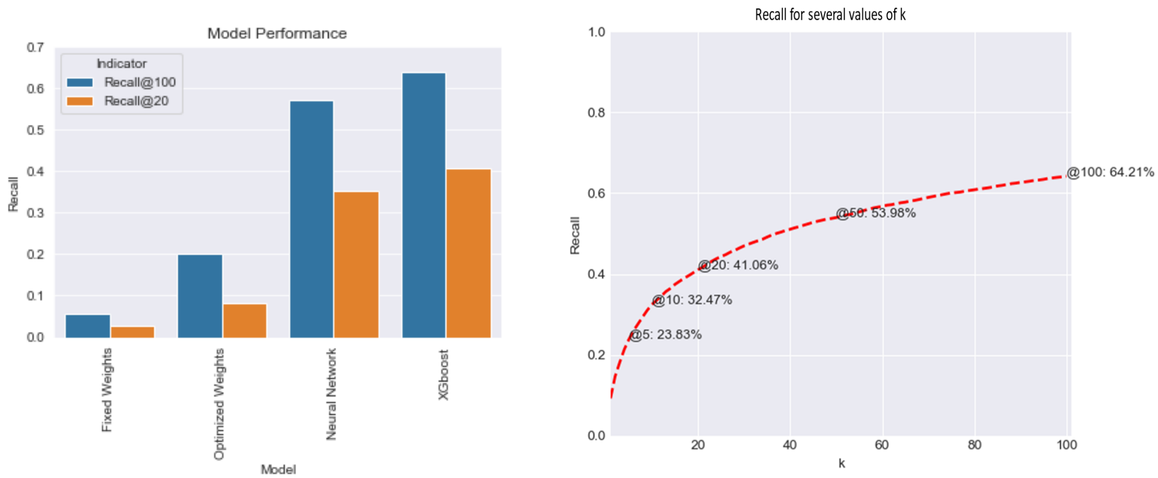


Figure 1: Performance on a test set of different recommender systems.

5%, is very low. The second recommender system considered uses the same variables as these used to build the matching score, but instead of giving them fixed weights, it optimizes them so as to best predict return to employment. This leads to improvements, but the recall@100 is still modest, remaining below 20%. The last two recommender systems consider a broader set of variables. The first of the last two systems, based on neural networks, follows the method described in Section 2.2. The last one uses another machine learning method based on ensembling and leverages variables that explicitly describe interactions between the variables characterizing job supply and job seekers (*e.g.* the distance between a job seeker and an establishment). Both recommender systems perform significantly better than the first two. The neural network achieves a recall@100 of about 57.5% and the last one achieves an even higher recall@100. The disadvantage of this last system, however, is its speed, especially when issuing recommendations. Training the neural network based model takes around an hour, and generating a set of recommendations for a given jobseeker with it around 0.07 seconds; those numbers are respectively 2 hours and 10 seconds for the last model.

The right panel of Figure 1 shows how the recall of the last model varies with the number of recommendations. For 5 recommendations, the proportion is as large as almost 20%. As shown in the figure, the proportion increases progressively when the number of recommendations increases.

3 In-the-lab comparison of different recommender systems

In our recommender system, we consider the score which maximizes the chances to find a job whereas the preference-based system rather uses a score consistent with jobseekers' search criteria. A legitimate question is how do they compare if we consider either best recommendation $\Pi^*_{i,\cdot}$ (for the preference-based system) and $S^*_{i,\cdot}$ (for ours).

An additional layer of complication is that the score of our recommender system has to be calibrated using a set of applications to be able to be interpreted as a probability to match on a job offer. Rather than integrating this constraint in the output of the recommender system, we use a two steps approach. Here, as detailed in Section 2.2.2, we leverage the fact that we have access to the set of all applications, failed or not, of the job seekers made through the PES. We can then calibrate our score $S^*_{i,\cdot}$ and compare the recommendations with respect to their potentialities in terms of matching probability as well as with respect to the job seekers' search criteria adequacy.

For the sake of simplicity and tractability, we focus hereafter on the subsample of jobseekers and offers whose main sector is transportation and logistic in the French Rhône-Alpes region from ISO weeks 1 to 48 of 2019. It contains 60,299 jobseekers and 18,873 offers.

3.1 Performance measures and comparisons

We see the recommendation of a vacancy j to a jobseeker i as a lottery $\mathcal{L}$ characterized by its chance of success $p(i, j)$ and its gain $\Pi_w(i, j)$. In this section we first consider a single recommendation coming from our preference-based recommendation system and from our prediction-based recommendation system. We compare the two associated lotteries and examine the changes in the probability of success and the preference score as well as its components when switching from one lottery to the other.

Two central lessons emerge. The first is that there is much more relevant information that can be used to make recommendations than the usual criteria listed in the table. The second is that the information contained in the usual criteria is mobilized in an overly restrictive way. The very strong weights placed on certain characteristics such as skills lead to the exclusion of jobs that have characteristics that would be preferred by job seekers. Among others, this underlines the difficulty of specifying a preference-based system.

To explore the changes in matching probabilities and preference score as well as its components when switching from one recommender system to another, we consider only one recommendation ($k = 1$), $\mathcal{J}_1^*(i)$.[2]

Figure 2 provides a first graphical evidence of the implications of the switch. Each graph presents first two dimensional histograms, represented as heat-maps, of the joint distribution of $p(i, j)$ (on the "x" axis) and the preference score $\Pi_w(i, j)$ (on the "y" axis). Figure 2 (a) shows the distribution of the PES's best recommendation. In this figure, almost all the lotteries have a low probability and a preference score close to 1. Figure 2 (b) presents the distribution for the prediction-based recommendation. It shows a large shift of the distribution toward larger probabilities but also a substantial reduction in the preference score. Figure 2 (c) displays the distribution of the changes in preference scores and matching probabilities associated with switching from the preference based recommender system to the matching probability one. The distribution roughly has three modes. One is associated with a large gain in probabilities and a reduced loss in preference-based score. The second and largest is characterized by a larger loss in preference score and a more diverse gain in the matching probability. The last and smallest mode also has a large gain in matching probability but a more severe loss in the preference-based score. Note that recommendations from the two systems coincide for some individuals.

[2] We could focus as well on a sequence of recommendations. The individual performance measures can be easily generalized to a set $\mathcal{J}_k^*(i)$ of k recommendations. We define first the probability of a match on one of the recommended vacancies and then the average of any characteristic of the match. The probability of a match on the recommended set $\mathcal{J}_k^*$ of k recommendations can be simply written as

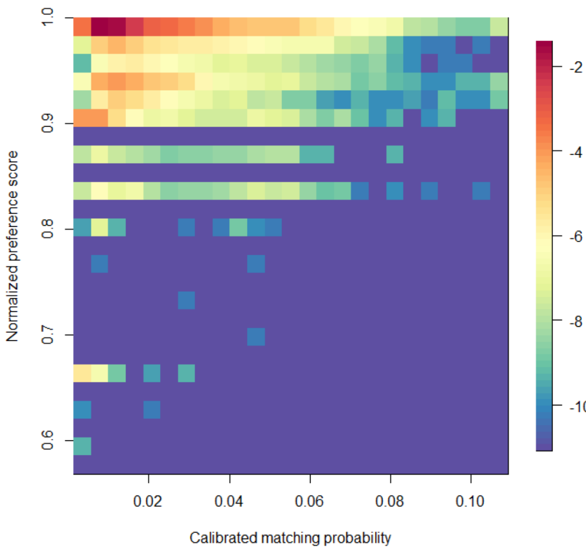
$$P(i, \mathcal{J}_k^*(i)) = 1 - \prod_{j \in \mathcal{J}_k^*(i)} (1 - p(i, j))$$

The expected characteristics X of the matched offer are

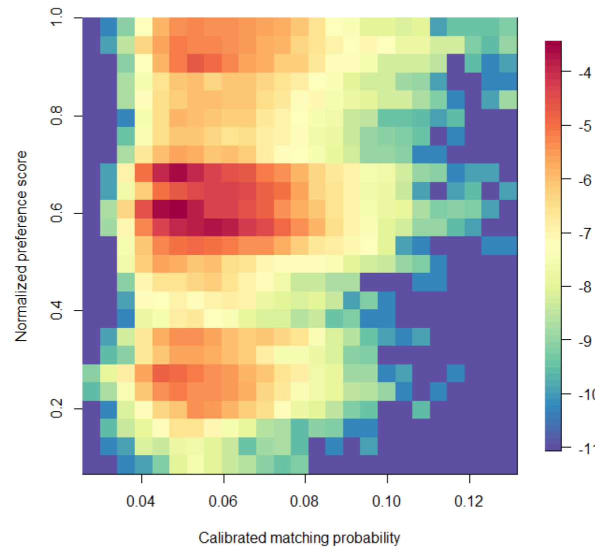
$$X_i(\mathcal{J}_k^*(i)) = \frac{1}{k} \sum_{j \in \mathcal{J}_k^*(i)} x_{i,j} p(i, j)$$

This can be used to define the average of the preference-score $\Pi_w(i, j)$ over the set of recommendations $\mathcal{J}_k^*$, or the average of the sub-scores or also the average of each of their components.

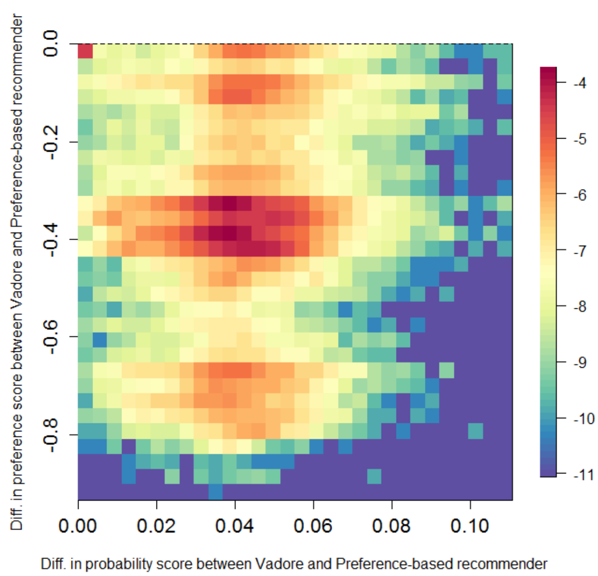
Figure 2: Distribution of preference score and matching probabilities for the two recommender systems.



(a) Preference based recommender



(b) Prediction based recommender



(c) Switch between preference based and prediction based recommender

Notes: Log-frequencies on a sample of 60,299 jobseekers.
Upper panel: Histograms of the best recommendations for a preference-based (left) and prediction-based (right) recommender system on the matching probability and the preference score.
Lower panel: Histograms of the changes in preference score and matching probability associated with switching from preference-based to prediction-based recommender system.

Tables 4 and 5 detail the magnitude of gains in the probability of finding a job and losses in the preference score. They also explore the nature of these losses. Each column of the table correspond to a quintile of the distribution of the variation of a score. Table 4 uses the quintiles of the variation of the preference score Π_w and table 5 uses those of the probability of finding a job $p(i, j)$. Each line corresponds to the average of a criterion on each of these quintiles and the related 95% confidence intervals. The first row corresponds to the variation of the global preference criterion and each of the following rows corresponds either to one of the two sub-scores or to one of the components entering the definition of this sub-score. Finally, the last line gives the average of the variation of the probability of finding a job.

A first joint lesson from the two tables 4 and 5 is the independence between the gains in matching probability and the variations in the preference score. Indeed, the variation of the matching probability in the Table 4 is uniformly 0.04 across the different quintiles of the preference score variation distribution. In the same way the decrease of the preference score is very homogeneous through the quintiles of the distribution of the variation of the matching probability in the Table 5. This means that it is not because a jobseeker could see his chances of returning to work improve with the offers from the prediction-based algorithm that he should consent to a greater loss of preference score. Therefore, the columns of the table describing the decomposition of the preference score according to its components are very close to each other.

The average decrease in the preference score is very substantial, about -0.38. The increase in the chances of returning to work is due to the abandonment of many job selection criteria. The criteria which are the most affected are skills and driver's license for the sub-criterion on job seeker profile and type of job sought, salary and type of contract.

Another lesson from Table 4 is that there are jobseekers who have to make very substantial losses in their preference score. In the first quintile of the distribution, the loss in preference score is on average 0.67. For a second group, gathering quintiles 2 to 4, the decrease is between -0.3 and - 0.4. Finally, for the first quintile, the loss of score is on average limited, of the order of -0.09. The switch from the preference-based recommendation system to the prediction-based recommender system leads for all quintiles to losses on the criterion of salary (line 11 of the table). The increase in the chances of returning to work brought by the prediction-based recommender system systematically includes taking jobs that pay less than the reservation wage. It also systematically involves taking a shorter contract than the one sought. The extent of the loss in preference score from one quintile to the next includes changing jobs, working different hours than desired, and applying for a job that does not match one's skills. To put it in a nutshell, the algorithm does not seem to perform miracles:

to maximize the chances of returning to work, it implies accepting less well-paid, more precarious jobs, changing jobs, accepting atypical schedules, and not exploiting fully one's skills.

The preference score that we consider is in fact a mix between a score describing who the job seeker is, the profile score, and what he is looking for (the search score). Figure A2 describes the relationship between the variation of the matching probability and the variation of each of these scores. It also illustrates the Table 5 by showing that the loss in the search score is less important than for the profile score. It worth mentioning that Figure A2 shows that some jobseekers gain either on the search or profile scores when the prediction based recommender system is used.

Table 4: Changes in scores and their components by quintiles of the change in preference score

	$Q1(\Delta\Pi_w)$	$Q2(\Delta\Pi_w)$	$Q3(\Delta\Pi_w)$	$Q4(\Delta\Pi_w)$	$Q5(\Delta\Pi_w)$
$\Delta\Pi_w$	-0.671	-0.43	-0.381	-0.316	-0.087
	[-0.673,-0.67]	[-0.431,-0.43]	[-0.381,-0.38]	[-0.316,-0.315]	[-0.0877,-0.0854]
$\Delta\Pi_{w,Profile}$	-0.72	-0.716	-0.71	-0.64	-0.102
	[-0.722,-0.718]	[-0.719,-0.713]	[-0.713,-0.708]	[-0.642,-0.637]	[-0.105,-0.0992]
$\Delta(Skills)$	-0.962	-0.936	-0.943	-0.887	-0.128
	[-0.964,-0.959]	[-0.94,-0.932]	[-0.947,-0.939]	[-0.891,-0.883]	[-0.132,-0.124]
$\Delta(Diploma)$	-0.036	-0.035	-0.019	-0.01	-0.011
	[-0.0395,-0.0319]	[-0.0384,-0.0316]	[-0.0217,-0.0164]	[-0.0119,-0.0075]	[-0.0126,-0.0085]
$\Delta(Language)$	-0.052	-0.028	-0.039	-0.013	-0.021
	[-0.0565,-0.0481]	[-0.031,-0.0251]	[-0.043,-0.0353]	[-0.0153,-0.0098]	[-0.0246,-0.0182]
$\Delta(Driv.\ License)$	-0.382	-0.558	-0.429	-0.06	-0.111
	[-0.391,-0.373]	[-0.567,-0.549]	[-0.438,-0.419]	[-0.0644,-0.0548]	[-0.117,-0.105]
$\Delta(Experience)$	0.005	-0.042	-0.027	-0.003	-0.008
	[0.0001,0.0105]	[-0.046,-0.0371]	[-0.0325,-0.0218]	[-0.006,0.0001]	[-0.0116,-0.0044]
$\Delta\Pi_{w,Search}$	-0.615	-0.204	-0.113	-0.036	-0.075
	[-0.618,-0.612]	[-0.206,-0.201]	[-0.116,-0.111]	[-0.0376,-0.0336]	[-0.0771,-0.0732]
$\Delta(Occupation)$	-0.861	-0.065	-0.048	-0.023	0.02
	[-0.867,-0.855]	[-0.0689,-0.0601]	[-0.0513,-0.0437]	[-0.0259,-0.0195]	[0.0175,0.0228]
$\Delta(Hours)$	-0.25	-0.182	-0.083	-0.025	-0.168
	[-0.258,-0.242]	[-0.189,-0.174]	[-0.0882,-0.0768]	[-0.0291,-0.0211]	[-0.175,-0.161]
$\Delta(Wage)$	-0.557	-0.867	-0.447	-0.184	-0.508
	[-0.566,-0.548]	[-0.873,-0.86]	[-0.456,-0.437]	[-0.191,-0.176]	[-0.517,-0.499]
$\Delta(Mobility)$	-0.043	-0.108	-0.015	0.035	-0.052
	[-0.0505,-0.0346]	[-0.114,-0.103]	[-0.0187,-0.0121]	[0.0293,0.0398]	[-0.0573,-0.0473]
$\Delta(Contract)$	-0.262	-0.396	-0.107	-0.333	-0.344
	[-0.273,-0.252]	[-0.406,-0.385]	[-0.116,-0.0981]	[-0.343,-0.323]	[-0.353,-0.334]
$\Delta(p)$	0.04	0.04	0.041	0.041	0.042
	[0.0397,0.0404]	[0.0402,0.0407]	[0.0412,0.0417]	[0.0407,0.0412]	[0.042,0.0427]

Notes: Π_w corresponds to the preference score defined in (1), $\Pi_{w,Profile}$ and $\Pi_{w,Search}$ respectively to the profile and search preference sub-scores of Π_w defined in (2) and (3). The other reported variations are the different criteria composing these scores: skills, diploma, languages, driving license, year of experience, occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract. These criteria take value 1 if satisfied by the recommended offer. The coefficients reported are means of this variation in criteria among the population corresponding to the column quintile of $\Delta\Pi_w$. Confidence intervals are given at the 5% level.

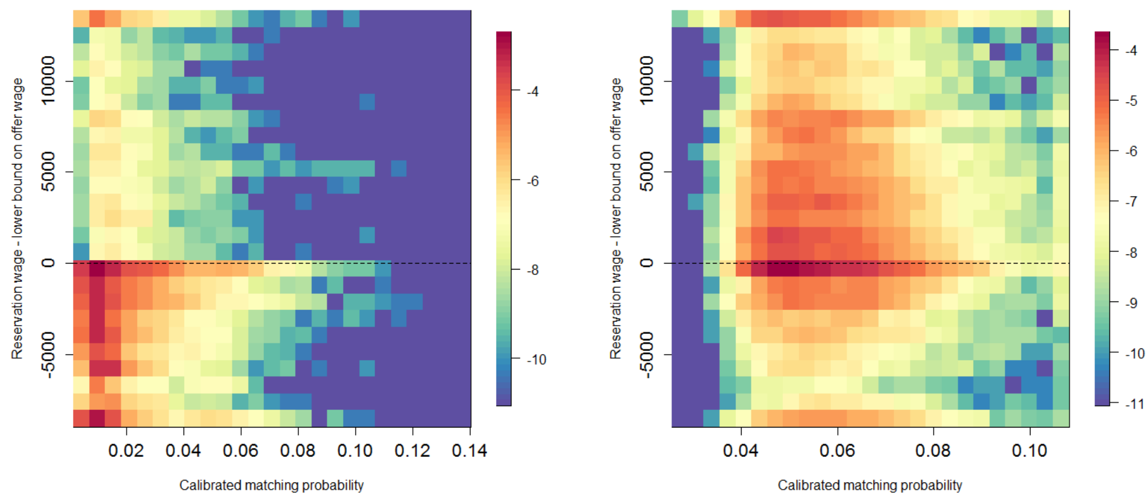
Table 5: Changes in scores and their components by quintiles of the change in calibrated probability

	$Q1(\Delta p)$	$Q2(\Delta p)$	$Q3(\Delta p)$	$Q4(\Delta p)$	$Q5(\Delta p)$
$\Delta\Pi_w$	-0.384	-0.395	-0.376	-0.375	-0.354
	[-0.388,-0.38]	[-0.399,-0.392]	[-0.38,-0.373]	[-0.379,-0.372]	[-0.357,-0.35]
$\Delta\Pi_{w,Profile}$	-0.573	-0.6	-0.575	-0.579	-0.562
	[-0.578,-0.568]	[-0.604,-0.595]	[-0.58,-0.57]	[-0.584,-0.573]	[-0.568,-0.557]
$\Delta(Skills)$	-0.777	-0.808	-0.77	-0.763	-0.737
	[-0.784,-0.77]	[-0.815,-0.802]	[-0.777,-0.764]	[-0.77,-0.756]	[-0.744,-0.73]
$\Delta(Diploma)$	-0.018	-0.034	-0.028	-0.015	-0.015
	[-0.021,-0.0149]	[-0.0374,-0.0301]	[-0.0313,-0.0252]	[-0.0173,-0.0129]	[-0.0173,-0.0126]
$\Delta(Language)$	-0.03	-0.028	-0.029	-0.026	-0.04
	[-0.0344,-0.0264]	[-0.0311,-0.025]	[-0.032,-0.0258]	[-0.0288,-0.0226]	[-0.0443,-0.0365]
$\Delta(Driv.license)$	-0.193	-0.238	-0.283	-0.41	-0.416
	[-0.201,-0.185]	[-0.246,-0.23]	[-0.291,-0.274]	[-0.419,-0.401]	[-0.424,-0.407]
$\Delta(Experience)$	-0.009	-0.01	-0.007	-0.017	-0.03
	[-0.015,-0.004]	[-0.015,-0.006]	[-0.011,-0.003]	[-0.022,-0.013]	[-0.035,-0.026]
$\Delta\Pi_{w,Search}$	-0.218	-0.219	-0.208	-0.21	-0.186
	[-0.223,-0.214]	[-0.223,-0.214]	[-0.212,-0.204]	[-0.215,-0.206]	[-0.191,-0.182]
$\Delta(Occupation)$	-0.22	-0.211	-0.189	-0.189	-0.166
	[-0.228,-0.213]	[-0.218,-0.204]	[-0.196,-0.182]	[-0.196,-0.182]	[-0.174,-0.158]
$\Delta(Hours)$	-0.097	-0.134	-0.17	-0.149	-0.157
	[-0.103,-0.0914]	[-0.141,-0.128]	[-0.177,-0.163]	[-0.156,-0.142]	[-0.164,-0.15]
$\Delta(Wage)$	-0.489	-0.524	-0.533	-0.527	-0.489
	[-0.498,-0.479]	[-0.533,-0.515]	[-0.542,-0.524]	[-0.536,-0.518]	[-0.498,-0.48]
$\Delta(Mobility)$	-0.081	-0.059	-0.036	-0.023	0.015
	[-0.0875,-0.0737]	[-0.0647,-0.0538]	[-0.0412,-0.0317]	[-0.0278,-0.0181]	[0.0092,0.0213]
$\Delta(Contract)$	-0.25	-0.352	-0.357	-0.24	-0.243
	[-0.26,-0.239]	[-0.362,-0.341]	[-0.367,-0.347]	[-0.25,-0.231]	[-0.252,-0.234]
$\Delta(p)$	0.019	0.0349	0.041	0.048	0.062
	[0.0191,0.0195]	[0.0349,0.0349]	[0.041,0.0411]	[0.0479,0.048]	[0.0618,0.0623]

Notes: Π_w corresponds to the preference score defined in (1), $\Pi_{w,Profile}$ and $\Pi_{w,Search}$ respectively to the profile and search preference sub-scores of Π_w defined in (2) and (3). The other reported variations are the different criteria composing these scores: skills, diploma, languages, driving license, year of experience, occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract. These criteria take value 1 if satisfied by the recommended offer. The coefficients reported are means of this variation in criteria among the population corresponding to the column quintile of Δp. Confidence intervals are given at the 5% level.

Figure 3 provides a better understanding of the role of wages in the recommender system. It documents the extend of the loss regarding the wage which is associated with the switch to the prediction-based algorithm. It shows again the recommendations based on preferences and those based on the predicted probability. On the x-axis is the predicted probability of return to employment. On the y-axis is shown the difference between the reservation wage and the offered wage. A negative value corresponds to an offered wage higher than the reservation wage while a positive value indicates an offered wage lower than the reservation wage. Figure 3 (a) clearly shows that most job offers selected by the preference based recommender system satisfy the wage criterion: reservation wage is below the offered wage and the probability of employment is small. Figure 3 (b) is associated with larger matching probabilities but, most of the vacancies correspond to a reservation wage equal to the wage offered in the vacancy. The important point in this figure is that the distribution also shows regions corresponding to a substantial reduction in the wage.

Figure 3: Effect of switching from one recommender system to another on the matching probability and wage adequacy with the job seeker's criteria.



(a) Preferences based recommender (b) Prediction recommender

Note: We use annual reservation wages as declared at the first interview with the caseworker at the PES.

Table 6 describes what are the characteristics of the jobweekers who benefit or are harmed by a change from a preference-based recommender system to a prediction-based one. The first column presents the estimates of the linear regression of $\Delta\Pi_w$ onto characteristics such as age categories, unemployment duration in the current spell, experience in the job sector, gender, number of children, qualification level, reservation wage and maximal mobility, firms and job densities. It shows that women can expect a decrease in their preference score as well as more experienced jobseekers. Column (1) of Table 6 shows the estimates of the average marginal effects (AME) in a logit model where the dependent variable indexes if one pertains to the fifth quintile of $\Delta(\Pi_w)$ rather than the first one. Thus, it indicates the characteristics of those who are likely to experience a small drop in their preference score rather than a large one. This complements the first columns and underlines that some categories such as specialized workers or technicians are better off then others after the change. Columns Δp and (2) perform a similar analysis on the changes in matching probabilities. This shows that long term unemployed, those with a high reservation wage, or low maximal mobility are more likely to have less gains in matching probabilities than others. This is closely related to the conclusions on the effect of the recommender system drawn previously in this section. The intermediate class age $[45, 55[$ seems also to benefit more than others from prediction-based recommendations. Columns (4) and (5) consider characteristics of those who benefit the more/less in terms of expected preference score for this best recommendation, *i.e.* the product of the preference score Π_w times the matching probability (see Section 3.2). This complements the previous analysis showing that women, older individuals (≥ 55), and those with high reservation wage or low maximal mobility are more likely to incur a loss in terms of expected preference score.

Table 6: Correlations between jobseekers characteristics and changes in scores and matching probability

	$\Delta\Pi_w$	(1)	Δp	(2)	(3)	(4)
Unempl. dur	0.0048	0.0061	-0.001*	-0.0431*	0.0021	-0.0015
Age $\leq$ 35	-0.0082*	-0.0321*	4e-04	0.0141	-0.0202	0.0044
Age [45,55[	-0.0135*	-0.0606*	0.0017*	0.0562*	-0.033	0.0089
Age $\geq$ 55	-0.0037	-0.02	8e-04*	0.0271	-0.027	0.0351*
Exper.	-6e-04*	-0.0022*	1e-04*	0.0035*	1e-04	-1e-04
Woman	-0.1459*	-0.3283*	-0.0016	0.0139	-0.2154*	0.1696*
Nb. Children	0.0013*	0.005*	-1e-04*	-0.0029*	0.003*	-0.0012*
Specialized worker	0.0149*	0.0488*	-0.0026*	-0.1216*	-0.0223	0.0048
Qualified worker	-0.0084	-0.0027	0.0032*	0.1262*	0.081*	-0.0236*
Sup. qualified worker	-0.0018	-0.0027	7e-04*	0.0174*	0.016*	-0.0079*
Unqualified employee	-4e-04	0.0019	5e-04*	0.0229*	0.016*	-0.0059*
Qualified employee	-0.004	0.0049	0.0025*	0.1436*	0.065*	-0.0149
Technician	0.0094	0.0717*	0.0011*	0.0789*	0.0208	0.0019
Worker manager	0.003	0.0442	0.0045*	0.2201*	0.1051*	-0.0084
Employee manager	-0.0023	0.0231	0.0045*	0.2386*	0.1205*	-0.0232*
Full time	-0.0073	0.0093	0.004*	0.216*	0.072*	-0.0062
Long term	-0.1098*	-0.2702*	0.0012	0.0993*	-0.1805*	0.1048*
Res. wage	0.1824*	0.2169*	-0.0252*	-0.6155*	-0.6039*	0.3155*
Res. mobility	-0.1834*	-0.3787*	9e-04	0.06	-0.258*	0.1915*
Firm density	0.0157*	0.0384*	-0.0032*	-0.151*	-0.0832*	0.0287*
Job density	-0.0126*	-0.0245*	-2e-04	-0.0163*	-0.0456*	0.0132*

Notes: 60,299 observations. Significance levels: 1% : *. The columns $\Delta\Pi_w$ and Δp are estimates in the linear regression of the variation in preference score $\Delta\Pi_w$ and matching probability Δp using the ML recommender rather than the one based on preference score on jobseekers characteristics. (1) are estimates of the AME in a logit model of pertaining to the fifth quintile of $\Delta(\Pi_w)$ rather than the first one, (2) are estimates of the Average marginal effects (AME) a logit model of pertaining to the fifth quintile of $\Delta(p)$ rather than the first one, (3) are estimates of the AME in a logit model of pertaining to the fifth quintile of $\Delta(\Pi_w p)$ rather than the first one. Finally column (4) are estimates of the AME in a logit model of whether $\Delta(\Pi_w p) < 0$.

3.2 Taking into account preferences and matching probability in the recommendations

So far, we have emphasized just two recommendation systems based on the scores. One based on the preference score $\Pi_w(i,j)$ and one based on the prediction score $p(i,j)$. An alternative is to consider the expected preference score, which is simply in the case of a single recommendation the product of the preference score and the probability score:[3]

$$E(i,j) = \Pi_w(i,j)p(i,j). \tag{7}$$

The results appear first on Figure 4. This figure is built following the same model as Figure 2. The top panel shows the histogram of the joint distribution of the matching probability and the preference score for the optimal recommendation based on the preference score (on the left - the histogram also appears in the same place in Figure 2) and for the optimal recommendation based on the matching probability (on the right). The lower panel shows the histogram of the joint distribution of changes in preference score and matching probability when switching from one recommendation system to the other. The differences and similarities with the figure 2 are striking. The upper panel shows a similar shift to the right of the histogram when switching from one system to another; it is a little less pronounced but hardly. Thus maximizing the product $p(i,j)\Pi_w(i,j)$ rather than $p(i,j)$ is not associated to much loss. On the other hand, both histograms remain at the top of the graphs. This means that in comparison with the results obtained when maximizing the matching probability, we do not observe such a general and important decrease of the preference score. Similarly, when we look at the distribution of the variations of the matching probabilities and the preference score (lower panel), the main change is that the mode of the joint distribution is much higher on the preference score support. This first examination tends to show that maximizing the product $p(i,j)\Pi_w(i,j)$ rather than $p(i,j)$ does not lead to a lesser performance in terms of chances of finding a job and has the advantage of limiting the decrease in the preference score.

Table 7 details the variation of the preference score. Like Table 4, it gives the variations of the score for the quintiles of the variation of the distribution of the preference score. Focusing on the first quintile (i.e. those who lose the most in terms of preference score) we notice first that the variation in the preference score is much more limited than the one observed in the table 4. While the

[3] We could as well define a similar quantity for a set $\mathcal{J}$ of J^* recommendations:

$$\begin{aligned} E(i,\mathcal{J}) =& p(i,1)\Pi_w(i,1) + (1-p(i,1))p(i,2)\Pi_w(i,2) \\ &+ \cdots + (1-p(i,1))\cdots(1-p(i,J^*-1))p(i,J^*)\Pi_w(i,J^*), \end{aligned}$$

which requires in addition that recommendations are sorted according to $\Pi(i,j)$ within $\mathcal{J}$.

average decrease in the quantile was -0.67, it is now -0.197. The variation of the matching probability is similar (0.04). Each of the two sub-scores - the one based on the job seeker's profile and the one based on his/her search parameters - decrease less. The main reasons for the limited decrease in these scores are the skill and occupation criteria, for which we do not observe losses large as the previous ones. The same substantial reductions are observed, however, on the search parameters such as wage, contract or hours. We also observe that the decreases in the other quintiles are quite limited while the gains in matching probability are less important than those observed in the table 4. The main lesson is that apparently there are a sufficiently large number of vacancies of different type at the top of the distribution of each individual to substantially increase the preference score without losing to much on the matching probability side.

Table 7: Changes in scores and their components by quintiles of the change in preference score for a recommender based on (7)

	$Q1(\Delta\Pi_w)$	$Q2(\Delta\Pi_w)$	$Q3(\Delta\Pi_w)$	$Q4(\Delta\Pi_w)$	$Q5(\Delta\Pi_w)$
$\Delta\Pi_w$	-0.197	-0.087	-0.055	-0.012	0
	[-0.198,-0.195]	[-0.0873,-0.0868]	[-0.0552,-0.0547]	[-0.0123,-0.0118]	[0,0]
$\Delta\Pi_{w,Profile}$	-0.236	-0.033	-0.0306	-0.0161	7.7e-05
	[-0.241,-0.231]	[-0.0344,-0.0317]	[-0.0316,-0.0295]	[-0.0169,-0.0154]	[-0.001,0.001]
$\Delta(Skills)$	-0.297	-0.017	-0.011	-0.002	0
	[-0.304,-0.289]	[-0.0182,-0.015]	[-0.0123,-0.0101]	[-0.0021,-0.0009]	[-0.0006,6e-05]
$\Delta(Diploma)$	-0.013	-0.009	-0.011	-0.005	0
	[-0.016,-0.0104]	[-0.0106,-0.007]	[-0.013,-0.009]	[-0.007,-0.003]	[-0.0004,0.0009]
$\Delta(Language)$	-0.032	-0.011	-0.007	0.004	0.006
	[-0.0349,-0.0282]	[-0.0128,-0.008]	[-0.009,-0.004]	[0.0022,0.006]	[0.0045,0.0078]
$\Delta(Driv.\ License)$	-0.303	-0.271	-0.266	-0.2	-0.003
	[-0.311,-0.295]	[-0.279,-0.263]	[-0.275,-0.258]	[-0.208,-0.193]	[-0.006,-0.001]
$\Delta(Experience)$	0.017	-0.011	-0.029	-0.013	0.001
	[0.0126,0.0217]	[-0.0142,-0.007]	[-0.0345,-0.0232]	[-0.0165,-0.008]	[-3e-05,0.003]
$\Delta\Pi_{w,Search}$	-0.171	-0.142	-0.087	-0.020	-0.0003
	[-0.174,-0.168]	[-0.143,-0.141]	[-0.0882,-0.086]	[-0.0203,-0.0191]	[-0.0005,-7e-05]
$\Delta(Occupation)$	-0.077	0.005	0.002	0.001	0
	[-0.0829,-0.0713]	[0.004,0.007]	[0.001,0.003]	[0.0004,0.0014]	[-7e-05,0.0003]
$\Delta(Hours)$	-0.291	-0.138	-0.062	-0.026	0.001
	[-0.3,-0.282]	[-0.144,-0.131]	[-0.0671,-0.0564]	[-0.0297,-0.0222]	[-0.0001,0.0024]
$\Delta(Wage)$	-0.558	-0.757	-0.553	-0.029	-0.009
	[-0.567,-0.548]	[-0.765,-0.75]	[-0.563,-0.544]	[-0.0326,-0.0244]	[-0.0113,-0.0069]
$\Delta(Mobility)$	-0.134	-0.168	-0.022	-0.004	0.005
	[-0.144,-0.125]	[-0.175,-0.162]	[-0.0254,-0.0189]	[-0.0064,-0.001]	[0.0034,0.0061]
$\Delta(Contract)$	-0.412	-0.466	-0.111	-0.501	0
	[-0.423,-0.4]	[-0.476,-0.455]	[-0.119,-0.103]	[-0.511,-0.49]	[0,0]
$\Delta(p)$	0.04	0.034	0.034	0.033	0.026
	[0.0395,0.0401]	[0.034,0.0345]	[0.034,0.035]	[0.0324,0.0329]	[0.0259,0.0266]

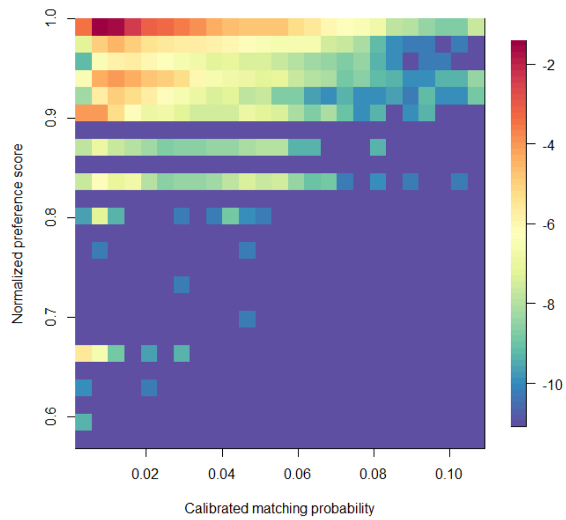
Notes: Π_w corresponds to the preference score defined in (1), $\Pi_{w,Profile}$ and $\Pi_{w,Search}$ respectively to the profile and search preference sub-scores of Π_w defined in (2) and (3). The other reported variations are the different criteria composing these scores: skills, diploma, languages, driving license, year of experience, occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract. These criteria take value 1 if satisfied by the recommended offer. The coefficients reported are means of this variation in criteria among the population corresponding to the column quintile of $\Delta\Pi_w$. Confidence intervals are given at the 5% level.

Table 8: Changes in scores and their components by quintiles of the change in calibrated probability for a recommender based on (7)

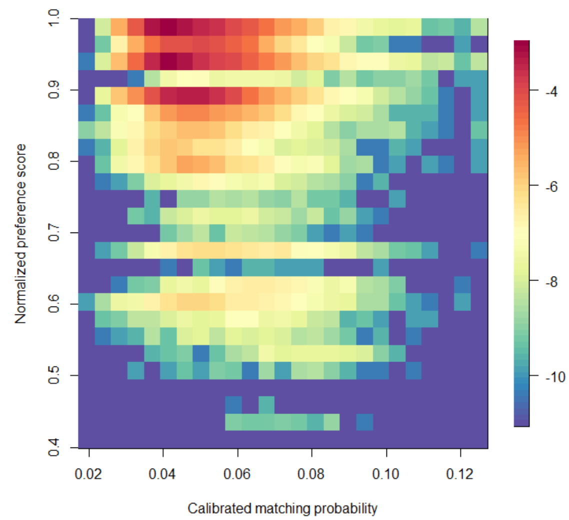
	$Q1(\Delta p)$	$Q2(\Delta p)$	$Q3(\Delta p)$	$Q4(\Delta p)$	$Q5(\Delta p)$
$\Delta\Pi_w$	-0.047	-0.063	-0.067	-0.073	-0.1
	[-0.048,-0.0458]	[-0.0646,-0.0619]	[-0.0685,-0.0658]	[-0.0747,-0.0717]	[-0.102,-0.0984]
$\Delta\Pi_{w,Profile}$	-0.020	-0.038	-0.050	-0.062	-0.146
	[-0.0214,-0.0185]	[-0.0404,-0.036]	[-0.0513,-0.0464]	[-0.065,-0.059]	[-0.151,-0.142]
$\Delta(Skills)$	-0.017	-0.037	-0.047	-0.058	-0.168
	[-0.0189,-0.0151]	[-0.0403,-0.0342]	[-0.05,-0.0432]	[-0.0616,-0.0542]	[-0.174,-0.162]
$\Delta(Diploma)$	-0.005	-0.005	-0.005	-0.007	-0.014
	[-0.0075,-0.0029]	[-0.0073,-0.0035]	[-0.007,-0.0037]	[-0.0088,-0.0055]	[-0.0163,-0.0119]
$\Delta(Language)$	0	-0.007	-0.004	-0.002	-0.025
	[-0.0018,0.0027]	[-0.0090,-0.0053]	[-0.006,-0.0027]	[-0.0047,-0.0002]	[-0.0288,-0.0221]
$\Delta(Driv.\ License)$	-0.098	-0.149	-0.203	-0.274	-0.32
	[-0.104,-0.0922]	[-0.156,-0.142]	[-0.21,-0.196]	[-0.282,-0.266]	[-0.328,-0.312]
$\Delta(Experience)$	-0.006	0	-0.005	-0.008	-0.014
	[-0.0110,-0.0015]	[-0.0047,0.0037]	[-0.0085,-0.0016]	[-0.0116,-0.0042]	[-0.0183,-0.0103]
$\Delta\Pi_{w,Search}$	-0.072	-0.089	-0.090	-0.094	-0.075
	[-0.0737,-0.0702]	[-0.0906,-0.0868]	[-0.0919,-0.0882]	[-0.0959,-0.0921]	[-0.0778,-0.073]
$\Delta(Occupation)$	-0.014	-0.019	-0.016	-0.019	0
	[-0.0164,-0.0121]	[-0.0219,-0.0169]	[-0.0185,-0.014]	[-0.0213,-0.0163]	[-0.0044,0.0035]
$\Delta(Hours)$	-0.033	-0.086	-0.127	-0.147	-0.123
	[-0.0364,-0.0291]	[-0.0914,-0.0801]	[-0.134,-0.121]	[-0.153,-0.14]	[-0.129,-0.116]
$\Delta(Wage)$	-0.268	-0.385	-0.423	-0.426	-0.405
	[-0.276,-0.259]	[-0.395,-0.376]	[-0.432,-0.413]	[-0.435,-0.417]	[-0.414,-0.396]
$\Delta(Mobility)$	-0.135	-0.096	-0.054	-0.037	-0.001
	[-0.142,-0.128]	[-0.102,-0.0903]	[-0.0595,-0.0495]	[-0.0418,-0.0323]	[-0.0067,0.0042]
$\Delta(Contract)$	-0.167	-0.293	-0.384	-0.352	-0.294
	[-0.176,-0.158]	[-0.303,-0.283]	[-0.393,-0.374]	[-0.362,-0.342]	[-0.304,-0.283]
$\Delta(p)$	0.011	0.027	0.034	0.04	0.055
	[0.0106,0.0109]	[0.0272,0.0273]	[0.0338,0.0339]	[0.0403,0.0404]	[0.0551,0.0555]

Notes: Π_w corresponds to the preference score defined in (1), $\Pi_{w,Profile}$ and $\Pi_{w,Search}$ respectively to the profile and search preference sub-scores of Π_w defined in (2) and (3). The other reported variations are the different criteria composing these scores: skills, diploma, languages, driving license, year of experience, occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract. These criteria take value 1 if satisfied by the recommended offer. The coefficients reported are means of this variation in criteria among the population corresponding to the column quintile of Δp. Confidence intervals are given at the 5% level.

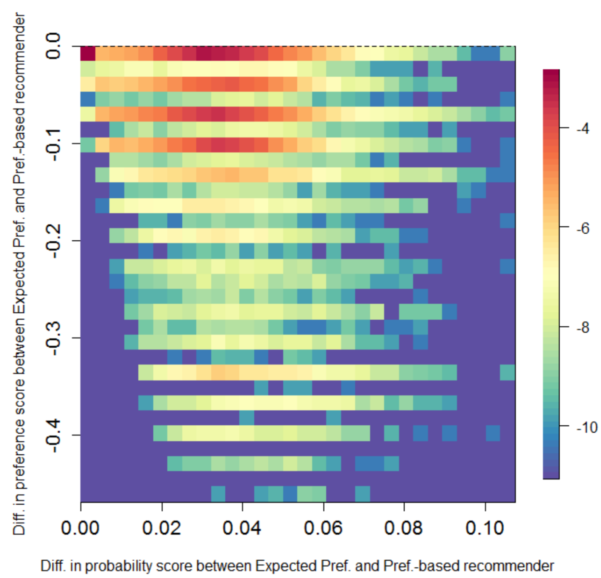
Figure 4: Distribution of preference score and matching probabilities for the preference-based and product of scores-based recommender systems.



(a) Preference based recommender



(b) Prediction based recommender



(c) Switch between preference based and prediction based recommender

Notes: Log-frequencies on a sample of 60,299 jobseekers.
Upper panel: Histograms of the best recommendations for a preference-based (left) and product of scores-based (based on (7) – right) recommender system on the matching probability and the preference score.
Lower panel: Histograms of the changes in preference score and matching probability associated with switching from preference-based to product of scores-based recommender system.

4 Congestion in Standard recommendation

The recommendation policy studied so far is defined at the individual level. Namely, after calibrating $p(i,j) := P(M^*_{i,j} = 1|S_{i,j}, C_{i,j} = 1)$, following Section 2.2.2, we consider the recommendation to a jobseeker i of the job ad with highest likelihood of leading to the signature of a contract, *i.e.* the ad indexed by $\text{argmax}_j\, p(i,j)$.

Yet a job ad comes with capacity constraints: a recruiter may only hire a single candidate on a given job opening.[4] Thus, recommending the same job ad, no matter how relevant it may be, to all jobseekers would be a mistake from the point of view of the PES. This strategy would induce a phenomenon of *congestion* at the population level, leaving both jobseekers (only one of which would be selected) and recruiters (the majority of which left unseen) unsatisfied.

The present section documents the phenomenon of congestion and establishes that the issue is not only theoretical. Although there are over 3 times more jobseekers than job ads in our dataset, top-1 recommendations addressed to them using the policy described so far cover less than 10% of job ads. Accordingly, efficient recommendation policies must be defined at the population level, taking congestion into account by design. We propose such a congestion-avoiding recommendation policy using tools from the computational optimal transport literature, and document the trade-offs it entails.

4.1 Individual-level recommendations generate congestion

We start by analyzing the congestion that would be generated by the "naive" recommendation system that consists in searching for each individual i the offer $j(i)$ maximizing his "instantaneous" chances of returning to work, ignoring competition. This algorithm is likely to recommend the same offer several times to different individuals and thus to generate competition between jobseekers. Each individual thus has competitors on the offer that is recommended to him, which are the set of i' such that $C(i) := \{i'|j(i') = j(i)\}$. We denote by $n^c(i)$, the number of competitors of jobseeker i. We can also rank each jobseeker i in the set of his competitors with respect to the individual probability $p(i,j)$. We thus define the rank of i as the rank of $p(i,j)$ in $C(i)$. Similarly, we can compute the matching probability *ex-post*, taking into

[4] Recruiters sometimes signal several job openings in a given job ad. The framework we propose easily handles this possibility - simply change the constraints accordingly and the proposed algorithm remains valid. Without loss of generality, we assume in the following that a job ad corresponds to a single opening.

account the competition, following what is done in the literature on directed search (see, *e.g.*, Wright et al., 2017). However, we consider here that the job applicants are examined according to their ranks in the set of competitors to which they belong.[5] This leads to the definition of the probability $p^c(i)$:

$$p^c(i) = \prod_{i' \in C(i) | r(i') < r(i)} \left(1 - p_{i',j(i)}\right) p_{i,j(i)}. \tag{8}$$

Figure 5 shows the characteristics of the competition associated with the standard recommender system. Panel 5 (a) shows the density of the number of competitors each jobseeker faces and panel (b) the distribution of jobseekers' ranks. The lower panel shows the probability distribution resulting from the prediction-based algorithm and once corrected for supply side competition (on the right). Finally, the table in panel 5 (d) presents some quantiles of the corresponding distributions. The number of competitors on an offer can be very important: 50% of the jobseekers have at least 229 competitors and 10% have more than 1049. The distribution of the rankings also illustrates the strong competition very well: 90% of the jobseekers have at least 8 competitors ranked higher than them and half have at least 90. This results in collapsing the probability of matching: while the probability of matching is above 5.6 percent for at least 50 percent of jobseekers, it is less than 1 in 1000 for 50 percent of jobseekers. Panel 5 (c) is also particularly eloquent since it shows the distributions of the matching probabilities before and after adjusting for competition. The two distributions are perfectly split, the distribution taking into account the competition being strongly shifted to the left and more strongly concentrated.

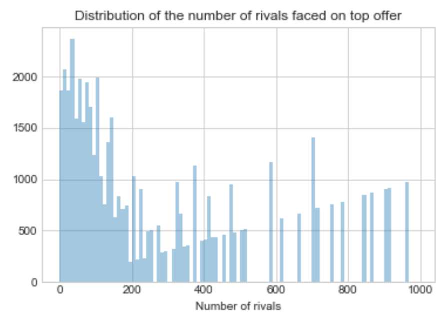
4.2 Congestion-avoiding recommendations using optimal transport

To reduce the competition between jobseekers on each vacancy, we use optimal transport theory. Indeed, if the competition that jobseekers face on their vacancy is important, it is also very heterogeneous. For each vacancy, we can determine the number of competitors and deduce the market share of each vacancy. This makes possible to examine the concentration of applications for the different vacancies. The Lorenz curve of these market shares illustrates the very strong concentration of the recommendations.[6] Figure 6 presents a

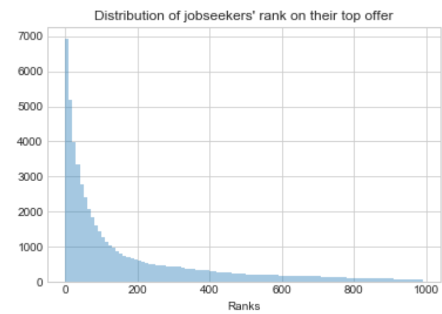
[5]The literature on directed search generally considers that the job applicants are taken in a random order and the matching probabilities are all identical.

[6]Remind that the Lorenz curve is obtained by classifying the vacancies according to the number of recommendations they receive. It then considers an increasing proportion of vacancies ranked in this way and plots the proportion of recommendations they receive against this proportion. The further the Lorenz curve deviates from the diagonal, the more concentrated the market is.

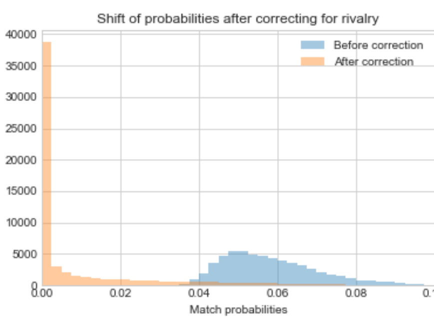
Figure 5: Competition for a job offer with the standard prediction-based recommendation system



(a) Density of $n^c(i)$



(b) Density of $r(i)$



(c) Density of $p(i,j)$ and $p^c(i)$

Quantiles	n^c	$p(i,j)$	$p^c(i)$	$r(i)$	$p(i,j) - p^c(i)$
0.10	31	0.044	6.7e-19	8	0.0233
0.25	78	0.049	2.0e-10	27	0.0423
0.50	229	0.056	0.0001	90	0.050
0.75	708	0.066	0.010	295	0.060
0.90	1049	0.076	0.036	632	0.069

(d) Quantiles

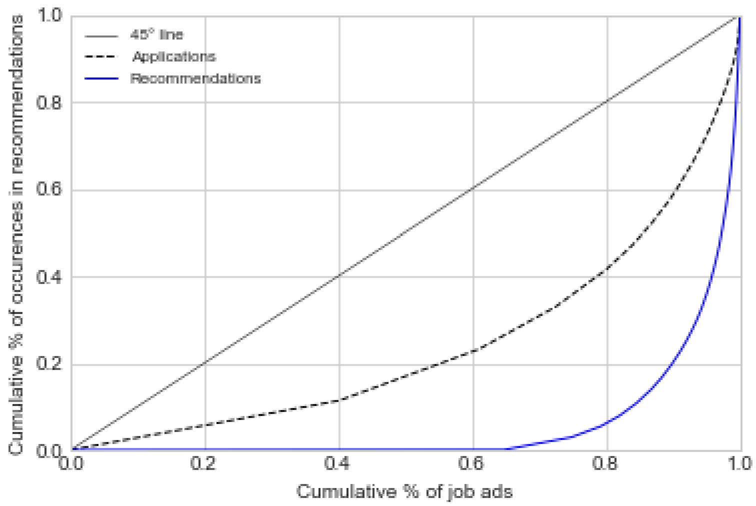
Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.
- n^c denotes the number of rivals faced on the top recommended offer;
- $p(i,j)$ is the calibrated matching probability of Section 2.2.2;
- $p^c(i)$ the *ex-post* matching probability defined in (8), taking rivalry into account;
- $r(i)$ denotes the jobseekers' rank on the top recommended offer.

Lorenz curve on the subset of jobseekers and offers which have been involved in an interview.[7] It shows the high concentration of recommendations resulting from the algorithm. For example, 80% of the vacancies receive less than 10% of the recommendations. Said another way, the 5% of vacancies receiving the most recommendations receive 60% of the total recommendations. We can also see in Figure 6 that the algorithm "increases" the concentration of

[7]To generate this curve, we considered all the vacancies in the MER file that had received at least one match and the jobseekers who were involved in these matches. For each of these jobseekers, if they had been matched with k vacancies in the file, we generated the k best vacancies from the algorithm. We thus generate a new (virtual) match file with as many matches as the initial file. It is on this file that we calculate the Lorenz curve of recommendations. It is thus directly comparable to the one obtained with the considered MER file.

applications. The same figure also shows the Lorenz curve for contacts between job seekers and vacancies. It is clear from the figure that the Lorenz curve of the recommendations from the algorithm is further away from the diagonal than the Lorenz curve of the observed matches. This result echoes the result obtained in another domain by Li et al. (2020) on the diversity of recruitment. The authors note in this paper that the diversity of recruitment is reduced when using recommendation algorithms.

Figure 6: Lorenz curve computed on "MER" matching file for job offers receiving at least one contact.



Notes: these results are based on the sub-sample of 3,652 jobseekers and 1,766 offers of the considered sector of transportation and logistic which have been involved in an interview (6,262 interviews have been recorded).
- "Applications" to the recorded interview, no matter the result.
- "Recommendations" corresponds to assignment plan based on the top recommendations of the prediction-based recommender system. When a jobseeker candidates on k offers, we consider the top k recommendations of the algorithm to compute the shares.

The goal of the optimal transport is to generate from our initial matching probabilities $p(i, j)$, a recommendation set, or assignment plan, that further distributes the applications on the vacancies. We briefly present the method in this section.

Denote N the number of jobseekers, and M the number of job ads, and by $\Gamma(N, M)$ the subset of the matrices $\mathcal{M}_{N,M}$ of size $N \times M$ satisfying

$$\Gamma(N, M) := \left\{ \gamma \in \mathcal{M}_{N,M} : \forall i, j,\ \gamma_{i,j} \geq 0,\ \sum_j \gamma_{i,j} = \frac{1}{N},\ \sum_i \gamma_{i,j} = \frac{1}{M} \right\}.$$

To each matrix $\gamma \in \Gamma(N,M)$ one can associate a discrete probability measure defining an assignment plan relating jobseekers to offers, hence $\Gamma(N,M)$ is the set of plans we consider as admissible. The first constraint in its definition is a positivity constraint, while the later two impose that the marginals of the discrete probability measure associated with γ satisfy the capacity constraints on the jobseekers and the offers. We consider the following linear program, where $\gamma \in \mathcal{M}_{N,M}$ has entries $\gamma_{i,j}$:

$$\begin{aligned} &\underset{\gamma}{\operatorname{argmax}} \ \sum_i \sum_j \gamma_{i,j} p(i,j) \\ &\text{s.t.,} \quad \gamma \in \Gamma(N,M). \end{aligned} \tag{9}$$

Given a assignment plan γ^* solution of problem (9), recommendations to jobseekers may be generated by taking in a deterministic way, for a given jobseeker i:

$$j(i) = \underset{j}{\operatorname{argmax}}\, \gamma^*_{i,\cdot}. \tag{10}$$

Alternatively, one can also draw randomly the ad's index according to a multinomial distribution with weights proportional to $\gamma^*_{i,\cdot}$. We call the latter way to recommend using the transportation plan the "probabilistic mode". Note that, due to the structure of the constraints, if recommendations to all jobseekers are issued with the probabilistic mode, any offer j will be seen on average N/M times.[8]

In real life scenarios, the linear optimization problem (9) under linear inequality constraints quickly reaches forbidding sizes as the algorithmic complexity of this problem scales is of order $\max(N,M)^{2.7}$ (see, *e.g.*, Weed, 2018; Peyré and Cuturi, 2019). We recall that in our case, γ is a $60,299 \times 18,873$ matrix. Fortunately, the structure of this problem has been studied in depth in the computational optimal transport literature. We turn to the following entropic relaxation, studied among others by Cuturi (2013); Galichon (2016); Galichon and Salanie (2011):

$$\underset{\gamma}{\operatorname{argmax}} \ \sum_i \sum_j \gamma_{i,j} p_{ij} - \varepsilon \gamma_{i,j} \log(\gamma_{i,j}) \tag{11}$$

$$\text{s.t.,} \ \gamma \in \Gamma(N,M) \tag{12}$$

As shown by Cuturi (2013), by inspection of the first order conditions, the regularized problem can efficiently be solved by the so-called Sinkhorn's algorithm, which is applicable for problems of the size of the one considered

[8] Rescaling weights for the multinomial draws, so that $\gamma_{i,\cdot}$ sums to 1, means multiplying through by N. For a given offer j, the number of times it is recommended would be the sum of N independent draws of a Bernoulli random variable with parameter $N\gamma_{ij}$, $i = 1, \dots N$. The expectation of that sum would be $\sum_i N\gamma_{ij} = N \times (1/M)$.

here. As shown in Proposition 4.1 of Peyré and Cuturi (2019), the solution of this regularized problem tends to a uniform coupling when $\varepsilon \to \infty$, and to the maximal entropy solution of the non-regularized problem when $\varepsilon \to 0$. ε may be considered as a "hyperparameter" of the approach. In practice, it should be tuned on a validation set in order to achieve a good trade-off between recommendation relevance, congestion, and computational considerations .

In the following, we obtain a recommendation plan γ by solving the regularized problem (11), and consider both deterministic and stochastic uses of the corresponding plans γ^*.[9]

4.3 Trade-offs between recommendation relevance and congestion

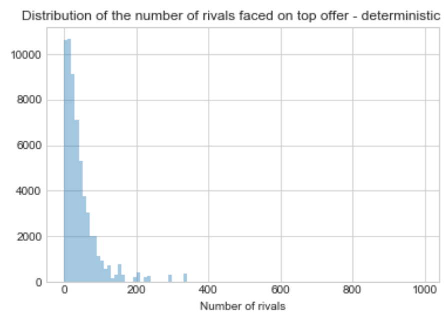
Figures 7 and 8 present the main results we obtain by applying optimal transport to deduce recommendations from the matching probabilities matrix. Figure 7 presents the results obtained when using optimal transport in deterministic mode, as defined in (10), and Figure 8 when using it in probabilistic mode. Each of the two figures is modeled on Figure 5. The impact of using optimal transport to make recommendations is visually striking. Overall, we observe that recommendations are more evenly distributed among vacancies and therefore generate less competition among jobseekers. The top panel of both figures shows a sharp reduction in the number of competitors each jobseeker faces, as well as a decrease in the rank of each jobseeker in the vacancy queue. The median number of competitors was 229 with the standard referral system, 29 with optimal transport in deterministic mode and 3 with optimal transport in probabilistic mode. A similar decrease is observed for the rank of jobseekers on their vacancies: the median rank was 90, it drops with optimal transport to 13 and 2 respectively.

The implication of this reduction in the number of competitors per vacancy can be seen directly on Figure 9, which presents the Lorenz curves of the shares of recommendations received by each vacancy. These curves are determined on the whole sample of jobseekers and offer in the transportation and logistic sector.[10] In this sample, the consistency of the recommendations with the standard recommendation system is particularly strong. More than 90% of the

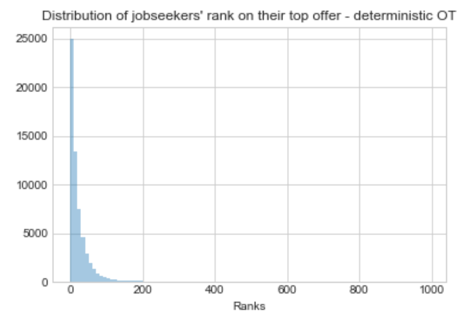
[9] To issue $k \geq 1$ recommendations for a jobseeker i, as is desirable in practice, one may solve problem (11), and proceed by either taking the top k indices in $\gamma_{i,\cdot}$, or draw repeatedly according to the multinomial (discarding duplicates until k distinct ads are drawn). Note that in the regularized setting, all entries of the optimal solution are strictly greater than 0 when $\epsilon > 0$; thus k non-degenerate top items may be found or drawn whenever k is lower than the number of job ads.

[10] Conversely, the Lorenz curves in Figure 6 were determined on the specific file of the MER connections– see footnote 7.

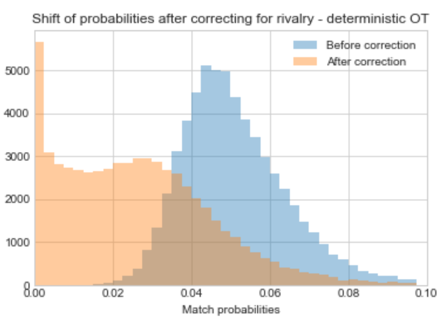
Figure 7: Competition for a job offer with the optimal transport recommendation system in deterministic mode



(a) Density of $n^c(i)$



(b) Density of $r(i)$



Density of $p(i,j)$ and $p^c(i)$

Quantiles	n^c	$p(i,j)$	$p^c(i)$	$r(i)$	$p(i,j) - p^c(i)$
0.10	6	0.035	0.002	2	0.002
0.25	14	0.041	0.010	5	0.009
0.50	29	0.049	0.024	13	0.023
0.75	55	0.059	0.038	28	0.038
0.90	95	0.069	0.051	54	0.048

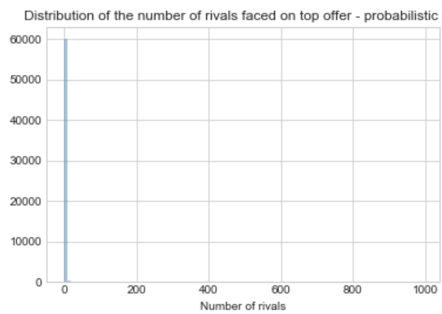
Quantiles

Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.
- n^c denotes the number of rivals faced on the top recommended offer;
- $p(i,j)$ is the calibrated matching probability of Section 2.2.2;
- $p^c(i)$ the *ex-post* matching probability defined in (8), taking rivalry into account;
- $r(i)$ denotes the jobseekers' rank on the top recommended offer.

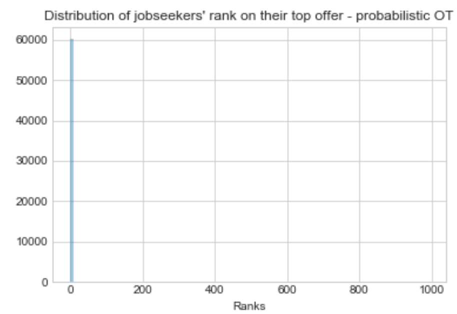
vacancies receive almost no recommendations. The congestion is substantially reduced with the recommendation system based on the optimal transport in deterministic mode. It is still important: 80% of the vacancies receive almost no recommendations. On the other hand, the use of the Optimal Transport in probabilistic mode leads to a very clear improvement. This is perfectly consistent with the remark in footnote 8 that balanced recommendation shares are obtained as a result of optimal transport in probabilistic mode.

On the other hand, there is a cost in terms of matching probability. As the examination of the global objective of (9) shows, the use of the optimal transport can only lead to a decrease in the instantaneous matching probabilities. This decrease is larger when optimal transport is used in probabilistic mode. As shown in the panels (d) of figures 5, 7, and 8, the median matching

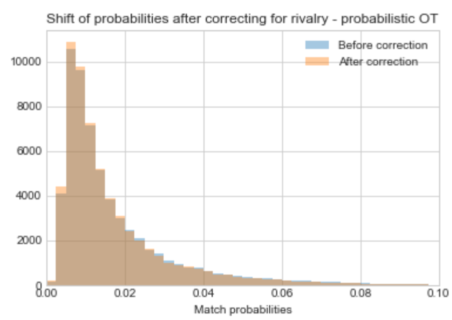
Figure 8: Competition for a job offer with the Optimal Transport recommendation system in probabilistic mode



(a) Density of $n^c(i)$



(b) Density of $r(i)$



Density of $p(i,j)$ and $p^c(i)$

q	n^c	$p(i,j)$	$p^c(i)$	$r(i)$	$p(i,j)-p^c(i)$
0.10	1	0.005	0.005	1	0.0000
0.25	2	0.007	0.007	1	0.0000
0.50	3	0.011	0.011	2	0.0001
0.75	4	0.021	0.020	3	0.0005
0.90	6	0.037	0.035	5	0.0019

Quantiles

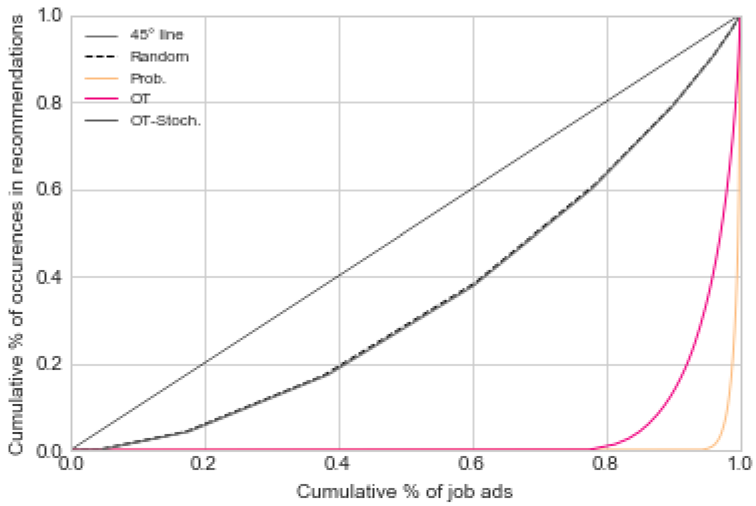
Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.
- n^c denotes the number of rivals faced on the top recommended offer;
- $p(i,j)$ is the calibrated matching probability of Section 2.2.2;
- $p^c(i)$ the *ex-post* matching probability defined in (8), taking rivalry into account;
- $r(i)$ denotes the jobseekers' rank on the top recommended offer.

probabilities are respectively 5.6, 4.9 and 1.1 for the standard recommendation system and those obtained by using optimal transport.

Figure 10 also illustrates this trade-off. It shows on the x-axis the matching probability and on the y-axis the log of the number of competitors that each job seeker encounters on the vacancy that is recommended to him. The mode of the distribution shifts downwards (fewer competitors) and to the left, decreasing the matching probabilities. This decrease is particularly clear when using optimal transport.

However, if the instantaneous matching probability decreases, the number of competitors also decreases. The adjustment to be made to go from the instantaneous probability to the effective probability taking into account the competition depends on the number of competitors (see (8)). Since there is

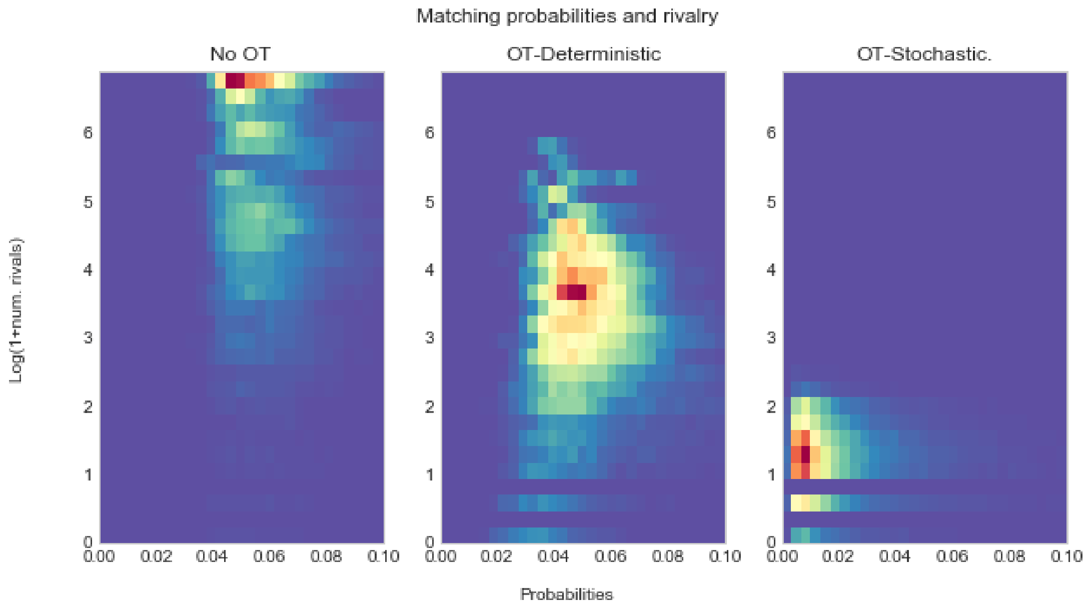
Figure 9: Lorenz curves for different recommender systems.



Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.
- "Random" corresponds to a random assignment plan between jobseekers and offers in this sector;
- "Prob." corresponds to assignment plan based on the top recommendations of the prediction-based recommender system;
- "OT" corresponds to the corrected assignment plan using optimal transport in deterministic mode (see (10)).
- "OT-Stoch" corresponds to the corrected assignment plan using optimal transport in probabilistic mode.

less competition when using optimal transport, the adjustment to be made is smaller. The matching probability after taking into account competition remains of course lower than the instantaneous probability, but it decreases much less than for the standard recommender system. As can be seen clearly in Figure 7, and even more clearly in Figure 8, the shift in the probability distribution when moving from the instantaneous probability to the competition-aware probability is much smaller than that observed in Figure 5. In the case of Figure 8, it is practically zero. The total effect, after taking competition into account, is therefore ambiguous. On the one hand, we systematically observe a decrease in matching probabilities when we switch from standard recommendations to recommendations using optimal transport. On the other hand, when we correct the matching probabilities to take into account the competition encountered by the job seekers, this decrease is no longer systematic. Figure A3 shows that we even observe an increase in probabilities. It is on the deterministic mode that the ex-post gains are the largest.

Figure 10: Joint distribution of rivalry and matching probabilities



Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.

Overall, the examination of the distributions reported in figures 5, 7 and 8 show that the largest ex-post matching probabilities are with optimal transport in deterministic mode. For example, the medians of the distributions of the ex-post matching probabilities with the standard recommendation system and the recommendation systems obtained via the optimal transport in probabilistic and deterministic mode are respectively 1/1000, 11/1000 and 24/1000.

5 Conclusion

In this paper we explore the properties of a recommendation algorithm trained within the French Public Employment Service and based on a rich data set. The goal of the algorithm is to identify, based on past matches, the profiles of the jobseekers and the offers which coincide the most. We first show how to interpret the matching score obtained from the machine learning algorithm in terms of matching probability. We then study the properties of the algorithm along two main directions. The first one is the adequacy of the recommendations made with the preferences of the job seekers. We compare two types of recommendations, those based on the matching probabilities of our algorithm and those based on the optimization of a preference score. We then describe the changes in the preference score and the matching probability when switching from one recommender system to the other. We show that the gains in matching probability are obtained at the cost of a considerable loss in the preference score. These initial results are not very exciting since they imply that the increase in the chances of returning to work, predicted on the basis of past transitions, is only obtained by accepting jobs that are less well paid than those to which one aspires, offering less stability, having shifted hours, that do not correspond to the desired occupation and that do not rely on skills. Nevertheless, we also show that when we try to maximize an objective which combines both the chances of returning to work and preferences, as is done in the literature in directed search (see, *e.g.*, Chade et al., 2017), we achieve gains in matching probability that remain significant. This also leads to losses in preference scores which are much more modest. A general lesson of this first "in the lab" part is that recommendation algorithms would benefit from being explicitly based on an objective on which job applications are based, such as expected utility.

In a first phase of our field experiment, we will randomly assign to each jobseeker a weight θ and then expose them to recommendations based on the maximization of $\Pi_w(i,j)^\theta p(i,j)^{1-\theta}$. This will allow us to identify the optimal θ parameter to consider for generating recommendations from the preference score and the matching probability.

In a second step of our "in the lab" analysis, we examine the issue of congestion. Without our recommendation algorithm, we document that congestion in the labor market is already high. Some jobs receive a very high share of recommendations, and others a very low share. We show that our algorithm tends to aggravate this phenomenon: it concentrates even more recommendations on certain offers. Using a formalism inspired from the directed search theory, we recompute the matching probabilities taking into account the competition between job seekers on the offers. We show that the concentration of job seekers on certain offers leads to a collapse of the matching probabilities. To solve this problem, our idea is to use optimal transport: instead of deriving recommendations from our probability matching matrix individually for each job seeker, we consider a collective objective which imposes that all offers be recommended as many times. The main result of using this objective is that it leads to recommendations with lower matching probabilities, thus requiring an apparent sacrifice from each individual. But the important point is that it also leads to a much lower number of competitors. Thus, once the competition between job seekers is taken into account, the chances of matching are higher with recommendations based on the optimal transport, despite the apparent sacrifice that must be made initially. However, the latter sacrifices might be an obstacle to the implementation of a recommendation system based on optimal transport.

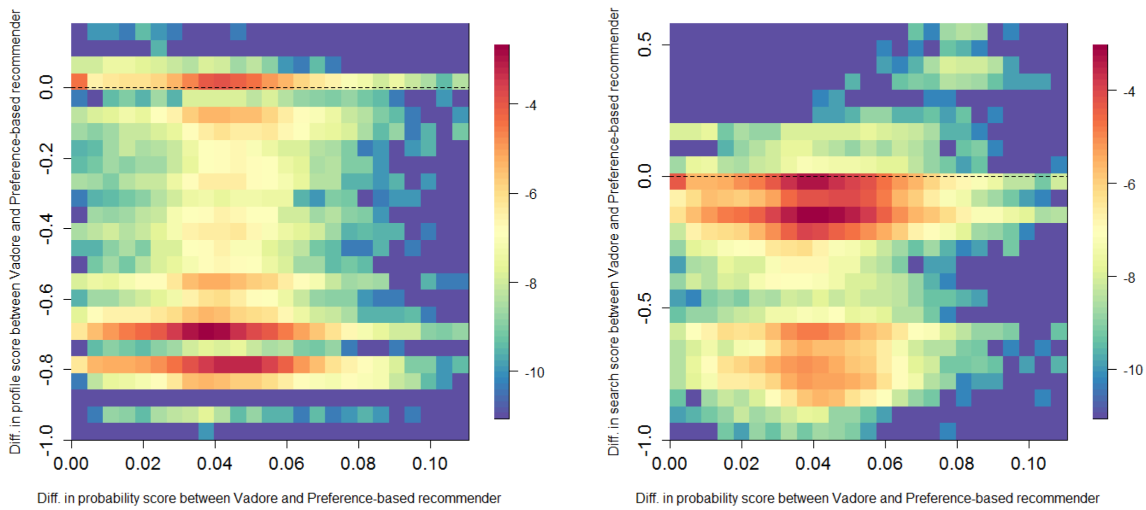
In a second phase of our field experiment, we will randomly assign micro markets to mobilize recommendations based on optimal transport or not.

A Supplemental Figures

Preference-based		Machine learning	
Jobseekers	Offers	Jobseekers	Offers
Skills	Skills	Skills (SVD, embedding)	Skill (SVD, embedding)
Diploma	Diploma	Diploma	Diploma
Languages	Languages		
Driver's licence	Driver's licence	Driver's licence	Driver's licence
Experience	Experience	Experience	Experience
Occupation (lv. 3)	Occupation (lv. 3)	Occupation (lv. 1, 2, 3)	Occupation (lv. 1, 2, 3)
Working hours	Working hours	Working hours	Working hours
Wage	Wage	Wage (several measures)	Wage (upper, lower bounds)
Location	Location	Location	Location
Geo. mobility		Geo. mobility	
Contrat type	Contract type	Contract type	Contract type
		Qualification	Qualification
		Soft skills	Soft skills
			Job description (text)
			Firm description (text)
			Contract type
			Contract duration
			Establishment size
			Establishment status
			Num. applications (ad)
			Num. applications (establishment)
			Num. days since posted
			Geo. soc.-dem. features
		Former occupation	
		Sex	
		Num. children	
		Search obligations	
		Job search type	
		Min. allowance status	
		Days unemployed	
		Age	
		Num. applications	
		Geo. soc.-dem. features	

Table 1: Information on offers and jobseekers that the preference-based and machine-learning based recommender systems leverage

Figure A1: Effect of switching from the preference based recommender system to prediction based on the sub preference scores

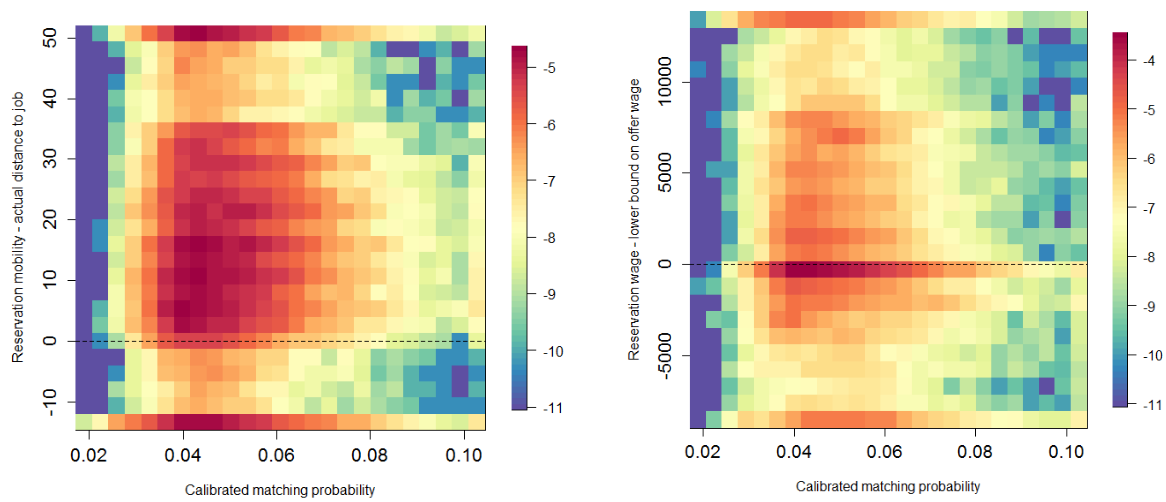


(a) On $\Pi_{w,Profile}$

(b) On $\Pi_{w,search}$

Notes:
- $\Pi_{w,Profile}$ is defined in (2) and aggregate criteria about skills, diploma, languages, driving license and year of experience.
- $\Pi_{w,search}$ is defined in (3) and and aggregate criteria about occupation, working hours, reservation wage, geographic mobility, duration, and the type of contract.

Figure A2: Effect of switching from a preference based system to a system based on (7) on the matching probability

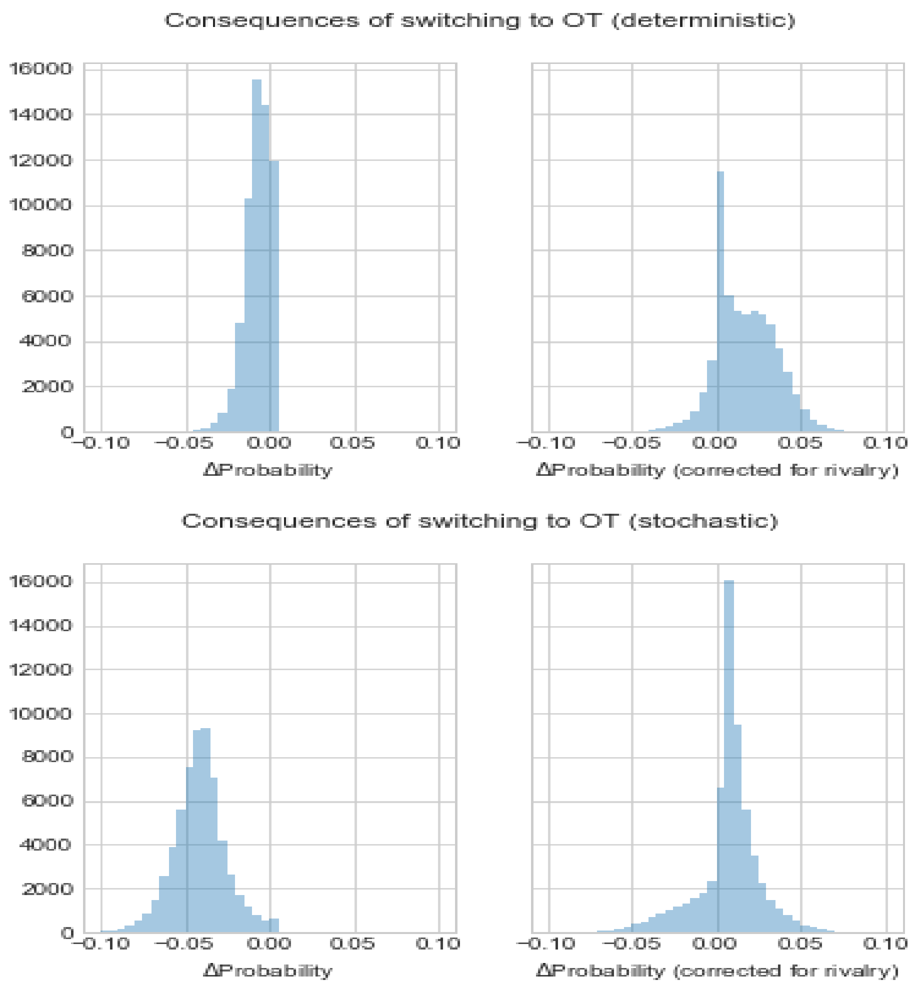


(a) On mobility criterion

(b) On wage criterion

Notes:
- Reservation mobility is the maximal distance at which jobseekers declare they are willing to accept an job offer;
- The lower bound on offer wage corresponds to the lower bound on the wage filled by the firm when posting the vacancy.

Figure A3: Changes in matching probabilities when switching from a standard prediction-based recommender system to a recommender using optimal transport.



Notes: these results are based on the 60,299 jobseekers and 18,873 offers in the transportation and logistic sector.

B Algorithm description

B.1 Feedforward neural networks

A single layer feed-forward neural network is a function from $\mathbb{R}^d$ to $\mathbb{R}^{d'}$ of the form $h(x) = \sigma(Wx)$, where σ is a nonlinear function applied point-wise (for instance, the sigmoid function), and $W \in \mathbb{R}^{d' \times d}$ is a weight matrix. In the case of the sigmoid and $d' = 1$, we recover a simple logistic predictor; when $d' > 1$, one may think of the output as d' logistic predictors side by side.

A k-layer feed-forward neural network stacks several single layer feed-forward networks, defining

$$h(x) = \sigma(W_k \sigma(W_{k-1} \ldots \sigma(W_1 x)))$$

where the W's are parameter matrices for the different layers. Each layer thus takes as input features the output of the previous one; each dimension of the layer's output combines its inputs linearly and passes them through a nonlinearity.

Various approximation theorems establish that, given arbitrary width or arbitrary depth, neural networks can approximate any well-behaved functions.

A crucial feature about neural networks is that using a set of usual nonlinearities, they remain differentiable (wrt the different weights W) by application of the chain rule. Automatic differentiation frameworks, such as *pytorch* (used for the implementation of the present algorithm), allow the training of such models (given a suitable loss function) with ease by stochastic gradient descent.

B.2 Overall network structure

As explained in 2.2, to incorporate domain knowledge, ϕ, ψ and A are given a block-wise structure, with three blocks corresponding to geography, to skills, and to other factors. In other words,

$$S_{ij} = \phi_g(x_{ig})^T A_g \psi_g(y_{jg}) + \phi_s(x_{is})^T A_s \psi_s(y_{js}) + \phi_o(x_{io})^T A_o \psi_o(y_{jo}),$$

where subscripts g, s and o correspond to "geography", "skills" and "other". Before a final step of end-to-end training, these three modules are trained separately to obtain warm-start values.

The margin parameter η is set to 1 for the separate training of the modules, and to 3 for the final end-to-end training.

In every module, negative examples are agressively subsampled. In all modules except for the pre-training of the "geographic" module, every epoch (*i.e.* mini-batch stochastic gradient descent pass on the training data), one negative example is uniformly sampled sampled to serve as contrast to every positive pair (amongst the offers available that week).

B.3 "Geography" module

The high-level inspiration of this module is drawn from Lian et al. (2014). A_g is set to the identity, ϕ_g to the identity. The contribution of this module to the overall score thus reduces to a scalar product of $\phi_g(x_{ig})$ and y_{jg}.

To compute y_{jg}, a grid of latitude-longitude-pairs of size 572 is first defined to cover the region's area [11]. Each dimension of $y_{jg} \in \mathbb{R}^{572}$ then corresponds to one of that grid's coordinates. The k-th dimension of y_{jg} is defined as:

$$y_{jgk} = \exp(-d(j,k)/10)$$

where $d(j,k)$ is the distance in kilometers from j's post code to the k-th latitude-longitude pair. Thus, y_{jgk} is close to 1 when j's establishment is geographically close to the k-th latitude-longitude pair, and tends to zero quickly for grid coordinates that are farther away.

ϕ_g takes an input $x_{ig} \in \mathbb{R}^{574}$: the representation of i on the same grid as used for y_g, concatenated with i's post code's raw latitude and longitude. x_{ig} is first mapped to a first layer of size $\mathbb{R}^{574}$ and composed with a tanh nonlinearity; then mapped to an embedding layer of size 572.

Training is done for 100 epochs with a batch size of 32, using the Adam optimizer and base learning rate 10^{-4}. Negative sampling selects items farther than the actual positive one. Furthermore, unlike in other modules, but in a customary fashion in the metric learning literature, 10 negative samples are chosen per positive one, and only the "hardest" one amongst those (*i.e.* the one for which the pair has lowest geographic score according to the current model state) contributes to the gradient.

B.4 "Skills" module

A "skill" 11272-200-100 module takes as input the one hot encoding of skills, with 1 hidden layer of size 200 (activation ReLU) and outputs a representation of size 100 (activation function tanh). The module is trained for 100 epochs (batch size 32) with the Adam optimizer and base learning rate 10^{-4}. The similarity matrix is kept to the identity during pre-training (only).

B.5 "Other" module

An "other" d-500-200 module takes as input the other descriptive features, with $d = 351$ for jobseekers and $d = 482$ for job ads, a first hidden layer of size 500 (activation ReLU) and an output representation of size 200 (activation function tanh). The module is trained for 100 epochs (batch size 256) with the Adam optimizer and base learning rate 10^{-4}.

[11] Resulting initially from a 25×25 flattened grid of geographic locations corresponding to quantiles in latitude and longitude respectively; duplicates coordinates are then discarded.

B.6 End to end training

The overall architecture is warm-started using the preliminary training of the above three modules. First, only ϕ_g, A_s, A_o are trained for 10 epochs with Adam optimizer and base learning rate 5×10^{-5}; then all modules are trained for 25 additional epochs with learning rate 10^{-5}. The batch size is 256 throughout end-to-end training.

C Additional Figures

References

Autor, D. H. (2001), 'Wiring the labor market', *The Journal of Economic Perspectives* **15**(1), 25–40.

Banerjee, A. and Chiplunkar, G. (2019), 'How important are matching frictions in the labour market? experimental & non-experimental evidence from a large indian firm', *Journal Article* .

Belot, M., Kircher, P. and Muller, P. (2018), 'How wage announcements affect job search-a field experiment'.

Belot, M., Kircher, P. and Muller, P. (2019), 'Providing advice to jobseekers at low cost: An experimental study on online advice', *The review of economic studies* **86**(4), 1411–1447.

Chade, H., Eeckhout, J. and Smith, L. (2017), 'Sorting through search and matching models in economics', *Journal of Economic Literature* **55**(2), 493–544.

Chen, K.-M., Hsieh, Y.-W. and Lin, M.-J. (2019), 'Prediction and congestion in two-sided markets: Economist versus machine matchmakers', *SSRN Electronic Journal* .

Chiappori, P.-A. and Salanié, B. (2016), 'The econometrics of matching models', *Journal of Economic Literature* **54**(3), 832–61.

Chilton, L. B., Horton, J. J., Miller, R. C. and Azenkot, S. (2010), Task search in a human computation market, *in* 'Proceedings of the ACM SIGKDD workshop on human computation', pp. 1–9.

Cuturi, M. (2013), Sinkhorn distances: Lightspeed computation of optimal transport, *in* 'Advances in Neural Information Processing Systems', p. 2292–2300.

Dupuy, A. and Galichon, A. (2014), 'Personality traits and the marriage market', *Journal of Political Economy* **122**(6), 1271–1319.

Galichon, A. (2016), *Optimal transport methods in economics*, Princeton University Press.

Galichon, A. and Salanie, B. (2011), 'Cupid's invisible hand: Social surplus and identification in matching models', *SSRN Electronic Journal* .

Gee, L. K. (2019), 'The more you know: information effects on job application rates in a large field experiment', *Management Science* **65**(5), 2077–2094.

Goodfellow, I., Bengio, Y. and Courville, A. (2016), *Deep Learning*, MIT Press. http://www.deeplearningbook.org.

Horton, J. J. (2017), 'The effects of algorithmic labor market recommendations: Evidence from a field experiment', *Journal of Labor Economics* **35**(2), 345–385.

Jenkins, S. P. et al. (1995), 'Easy estimation methods for discrete-time duration models', *Oxford bulletin of economics and statistics* **57**(1), 129–138.

Kässi, O. and Lehdonvirta, V. (2018), 'Online labour index: Measuring the online gig economy for policy and research', *Technological forecasting and social change* **137**, 241–248.

Kasy, M. and Abebe, R. (2021), Fairness, equality, and power in algorithmic decision-making, *in* 'Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency', pp. 576–586.

Lazear, E. P., Shaw, K. L. and Stanton, C. T. (2018), 'Who gets hired? the importance of competition among applicants', *Journal of Labor Economics* **36**(S1), S133–S181.

Li, D., Raymond, L. R. and Bergman, P. (2020), Hiring as exploration, Technical report, National Bureau of Economic Research.

Li, R., Ye, X., Zhou, H. and Zha, H. (2019), 'Learning to match via inverse optimal transport.', *J. Mach. Learn. Res.* **20**(80), 1–37.

Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E. and Rui, Y. (2014), Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation, *in* 'Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '14, Association for Computing Machinery, New York, NY, USA, p. 831–840.

Mas, A. and Pallais, A. (2017), 'Valuing alternative work arrangements', *American Economic Review* **107**(12), 3722–59.

Peyré, G. and Cuturi, M. (2019), 'Computational optimal transport', *Foundations and Trends® in Machine Learning* **11**(5-6), 355–607.

Weed, J. (2018), 'An explicit analysis of the entropic penalty in linear programming', *Preprint arXiv arXiv:1806.01879* .

Wright, R., Kircher, P., Julien, B. and Guerrieri, V. (2017), 'Directed search: A guided tour'.