A Task-Based Theory of Occupations with Multidimensional Heterogeneity

Sergio Ocampo

Western University

Occupations as bundles of tasks

An occupation is a bundle of tasks performed by a worker (Rosen, 1978)

- Professors research, teach, present, etc.
- Surgeons perform surgery, diagnose, etc.

Which tasks are bundled into an occupation? Which worker performs them?

- Workers' skills: Cognitive, manual, social, etc.
- Tasks' skill requirements:
  - Need high *cognitive skills* to be a Professor or a Surgeon
  - Need more *manual skills* to perform surgery than to teach

Skills and technology

Skills relevant for differences across workers:

- Levels of different skills affect wages
- Mismatch between worker's skills and tasks' skill requirements

Heckman & Scheinkman (1987); Autor, Levy & Murnane (2003); Spitz-Oener (2006); Poletaev & Robinson (2008); Kambourov & Manovskii (2009); Black & Spitz-Oener (2010); Yamaguchi (2012); Heckman & Kautz (2012)); Deming (2017); Guvenen, Kuruscu, Tanaka & Wiczer (2020); and Lise & Postel-Vinay (2020).

Skills and technology

Skills relevant for differences across workers:

- Levels of different skills affect wages
- Mismatch between worker's skills and tasks' skill requirements

Different skills affected differently by changes in technology:

- Skill-biased technical change
- Automatability of tasks

Heckman & Scheinkman (1987); Autor, Levy & Murnane (2003); Spitz-Oener (2006); Poletaev & Robinson (2008); Kambourov & Manovskii (2009); Black & Spitz-Oener (2010); Yamaguchi (2012); Heckman & Kautz (2012)); Deming (2017); Guvenen, Kuruscu, Tanaka & Wiczer (2020); and Lise & Postel-Vinay (2020).

Skills and technology

Skills relevant for differences across workers:

- Levels of different skills affect wages
- Mismatch between worker's skills and tasks' skill requirements

Different skills affected differently by changes in technology:

- Skill-biased technical change
- Automatability of tasks

Bundling of tasks into occupations both shapes effects and responds to changes

---

Heckman & Scheinkman (1987); Autor, Levy & Murnane (2003); Spitz-Oener (2006); Poletaev & Robinson (2008); Kambourov & Manovskii (2009); Black & Spitz-Oener (2010); Yamaguchi (2012); Heckman & Kautz (2012)); Deming (2017); Guvenen, Kuruscu, Tanaka & Wiczer (2020); and Lise & Postel-Vinay (2020).

Technology changes occupations

Tasks workers perform change $\longrightarrow$ Affects distribution of wages and employment

- Automation takes over some, *but not all*, tasks of an occupation
  - 50% of all tasks are currently automatable (McKinsey Global, 2017)
  - Less than 5% of occupations are fully automatable
- Occupations change directly by losing tasks (stockbrokers, phone operators) and indirectly by reassigning remaining tasks (manufacturing plant operators)
  - Occupations also respond to offshoring, IT, new tasks, worker training, etc.

A task-based theory of occupations

I **develop a framework** where boundaries of occupations are endogenous to ask:

- How are workers (wage, employment) affected by changes in occupations?
- What are the direction and effects of automation and technical change?

Framework: A **multidimensional assignment model** of tasks to workers

- Tractable despite multidimensional assignment (Villani, 2009; Lindenlaub, 2017)
- Highlights role of boundary tasks for wages and substitutability
- Endogenous response of occupations to technology

Roadmap

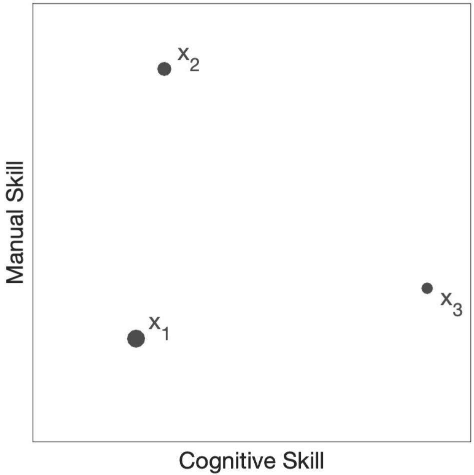
1. Task assignment model
2. Characterization of solution
   - Assignment
   - Productivity and wages
   - Elasticity of substitution across workers
3. Applications:
   - Directed Automation
   - Automation and unassigned tasks
   - Skill-biased technical change
   - Worker training

Model overview

- Consider an economy with a mass of workers
  - Workers differ in their skills: $x_n = (x_n^c, x_n^m, \ldots, x_n^s)$
  - There are N types of workers: $\mathcal{X} = \{x_1, \ldots, x_n, \ldots, x_N\}$
- Production combines output from a set of tasks $\mathcal{Y}$
  - Tasks differ in their skill requirements: $y = (y^c, y^m, \ldots, y^s)$
  - Task output depends on which worker performs the task (skill mismatch)

Objective: Assign tasks to workers to maximize production

Workers and skill endowments



Workers are characterized by a skill vector x_n

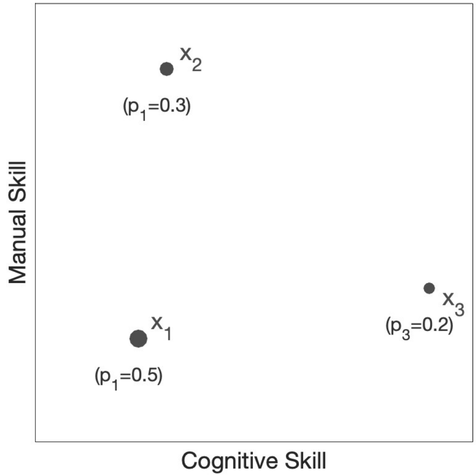
- Two skills: Cognitive and Manual
- A worker of type x_n is a pair:

$$x_n = (x_n^c, x_n^m)$$

- Finitely many types of workers:

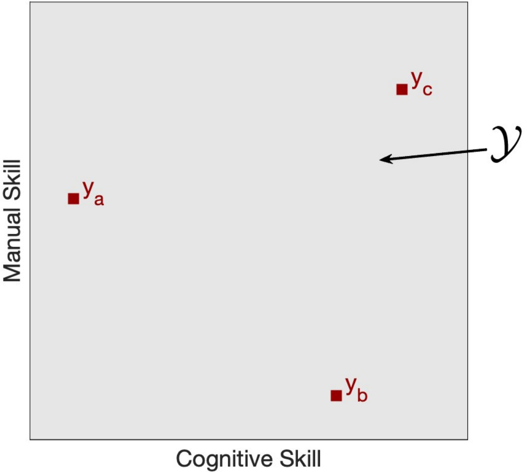
$$\mathcal{X} = \{x_1, \dots, x_n, \dots, x_N\}$$

Workers and skill endowments



- There is a mass p_n of workers of type x_n
- Each worker has one unit of time
- Workers have an outside option $\underline{w}(x)$
  - For simplicity, I set $\underline{w}(x) = \underline{w}$

Tasks and skill requirements



- Production requires set of tasks: $\mathcal{Y}$
  - $\mathcal{Y}$ assumed compact. Ex: $\mathcal{Y} = [0, 1]^2$
- A task (y) is characterized by skills:

$$y = (y^c, y^m)$$

- Tasks are continuously distributed on $\mathcal{Y}$
  - Distribution is G
- A task is completed in one unit of time

Production: Task output

Output depends on **match** between worker's skills and task's skill requirements:

- Function $q : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$ describes task output
- If worker x_n performs task y, task output is $q(x_n, y)$

Example:

$$\ln \mathrm{q}(\mathrm{x_n}, \mathrm{y}) = \underbrace{{a_x}' x_n}_{\text{Absolute Adv.}} - \underbrace{(x_n - y)' A (x_n - y)}_{\text{Worker/Task Mismatch}}$$

- Matrix A controls weights of skill mismatch, A is positive definite
- Linear term $\left({a_x}' x_n\right)$ affects productivity of workers across all tasks

---

Equiv. to tech. in Tinbergen (1956), Feenstra & Levinsohn (1995), Galicon (2016, Ch6) and Lindenlaub (2017).

Production

- Cobb-Douglas aggregate of output of all tasks

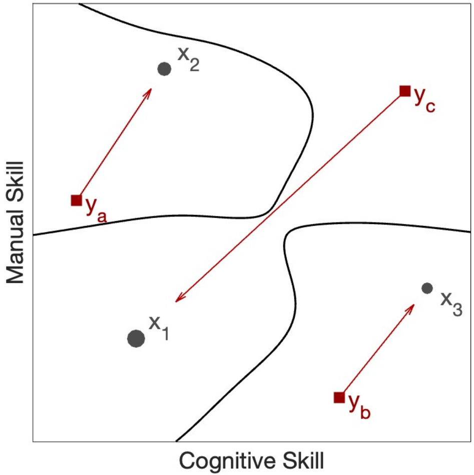
$$F(T) = \exp\left(\int_{\mathcal{Y}} \ln q\left(T(y), y\right) dG\right)$$

- All tasks must be performed to produce final good
- Production depends on assignment between tasks and workers

$$T : \mathcal{Y} \to \mathcal{X}$$

  - Task y is performed by worker $T(y) \in \mathcal{X} \equiv \{x_1, \ldots, x_n\}$

Assignment of tasks to workers

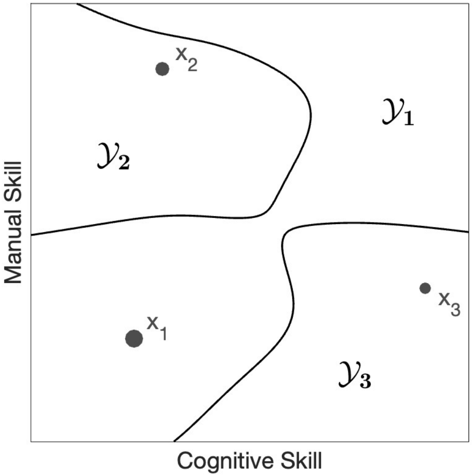


An assignment T matches tasks to workers

$\longleftarrow$ Example of an assignment
Not necessarily optimal

- T partitions task space
  - Assign each task to a worker

Assignment of tasks to workers



Def: Occupation ($\mathcal{Y}_n$)

- Tasks assigned to a worker (x_n)

$$\mathcal{Y}_n \equiv T^{-1}(x_n) = \{y \mid T(y) = x_n\}$$

Def: Demand for x_n (D_n)

- Time it takes to perform tasks in $\mathcal{Y}_n$

$$D_n \equiv \int_{\mathcal{Y}_n} dG$$

Optimal assignment

$$V(p_1, \dots, p_N) = \max_{T:\mathcal{Y}\to\mathcal{X}} \quad \exp\left(\int_{\mathcal{Y}} \ln q(T(y), y)\ dG\right) \qquad \text{s.t. } D_n(T) \leq p_n$$

Prop: If

i. $q(x, y) > 0$ for all (x, y) and $q(x, \cdot)$ is upper-semicontinuous

ii. q discriminates across workers: $\forall x_n \neq x_\ell$, $q(x_n, y) \neq q(x_\ell, y)$ G-a.e.

Then:

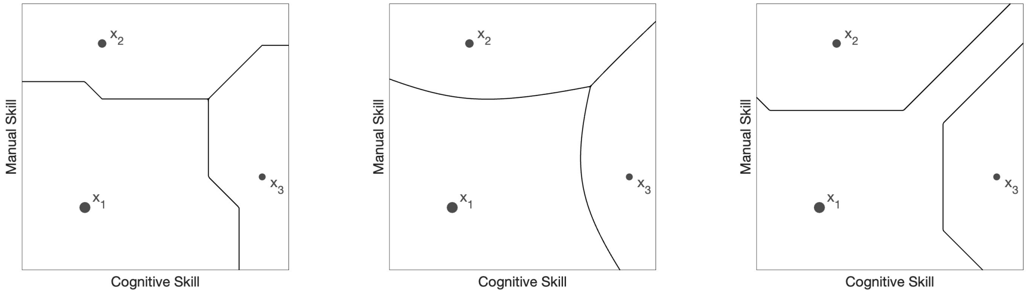
- There exists a G-unique solution $T^\star$
- There exists a unique $\lambda^\star \in \mathbb{R}^N$ with $\min \lambda_n^\star = 0$ s.t.:

$$T^\star(y) = \underset{x\in\mathcal{X}}{\operatorname{argmax}} \left\{\ln q(x, y) - \lambda_{n(x)}^\star\right\}$$

Roadmap

1. Task assignment model
2. Characterization of solution
   - Assignment
   - Productivity and wages
   - Elasticity of substitution across workers
3. Applications:
   - Directed Automation
   - Automation and unassigned tasks
   - Skill-biased technical change
   - Worker training

Task assignment and mismatch: $d(x,y) = \left(\sum_{i=1}^{d} |x_i - y_i|^p \right)^{1/p}$

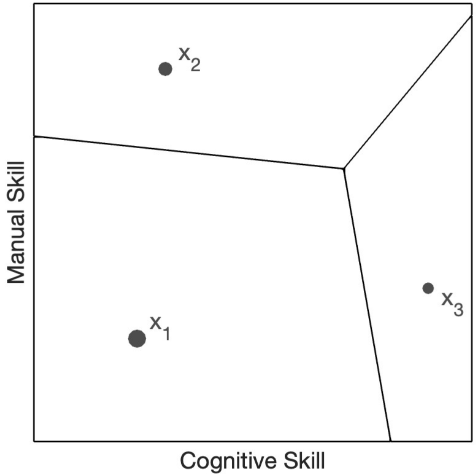


(a) Manhattan distance ($p = 1$) (b) Euclidean distance ($p = 2$) (c) Chebyshev distance ($p \to \infty$)

Min mismatch workers' and tasks'... subject to limited supply of workers:

$$\underbrace{D_n}_{\text{Demand for } x_n} \leq \underbrace{p_n}_{\text{Supply of } x_n}$$

Special case: $\ln q(x_n, y) = a_x' x_n - (x_n - y)' A (x_n - y)$

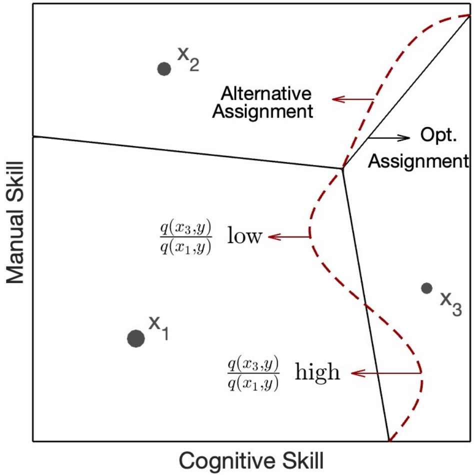


- Boundaries are hyperplanes:

$$0 = y' \underbrace{A(x_\ell - x_n)}_{\text{Normal Vector}} - \underbrace{\frac{1}{2}\left(x_\ell' A x_\ell - x_n' A x_n + a_x' (x_\ell - x_n) + \lambda_\ell^\star - \lambda_n^\star\right)}_{\text{Intercept}}$$

- Optimal assignment is a power diagram
  - Apply computational geometry tools

Special case: $\ln q(x_n, y) = a_x' x_n - (x_n - y)' A (x_n - y)$



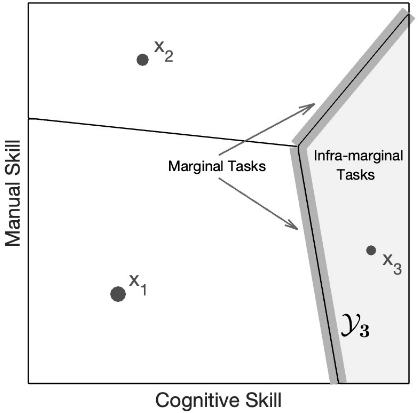
- Boundaries: constant ratio of output

$$\frac{q(x_n, y)}{q(x_\ell, y)} = e^{\lambda_n^\star - \lambda_\ell^\star}$$

- General feature for boundary tasks
  - Crucial for compensation of workers

Wages and marginal product are given by boundary tasks

$$w_n = \mathrm{MP}_n + \underline{w} \qquad \mathrm{MP}_n \equiv \frac{\partial V(p_1,\ldots,p_N)}{\partial p_n} = F(T^\star)\,\lambda_n^\star$$

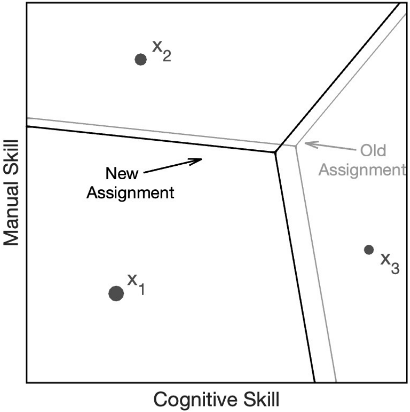


Change in output from task reassignment

- Consider an increase in p_3
- Optimal: assign tasks to x_3 along boundary
  - Lowest mismatch among unassigned tasks

Wages and marginal product are given by boundary tasks

$$w_n = \text{MP}_n + \underline{w} \qquad \text{MP}_n \equiv \frac{\partial V(p_1,\ldots,p_N)}{\partial p_n} = F(T^\star)\,\lambda_n^\star$$



MP_3 depends on productivity at boundary tasks

- Cascading by shifting boundaries
  - Output change: $\lambda_n^\star - \lambda_\ell^\star$ by boundary
  - $\lambda_3^\star$ captures cumulative gain
  - MP_3: increase of $100 \cdot \lambda_3^\star\%$ in output
- $\lambda^\star$ reveals ranking of workers
  - Productivity relative to lowest paid worker

Wage differentials and boundary tasks

Let $x_n \neq x_\ell$ and $y_n \in \partial\mathcal{Y}_n$ and $y_\ell \in \partial\mathcal{Y}_\ell$ be boundary tasks, then:

$$\underbrace{\lambda_n^\star - \lambda_\ell^\star}_{\text{diff. of multipliers}} = \underbrace{\ln q\left(x_n, y_n\right) - \ln q\left(x_\ell, y_\ell\right)}_{\text{(log) diff. of output in boundary task } y}$$

- Workers are compensated for differences at the margin:
  - If x_n produces 20% more than x_ℓ, then x_n receives 20% more of total output
- Wage relationship to skills (x) depends on boundary tasks (y)

Wage differentials: Quadratic production

$$\lambda_n^\star = \underbrace{a_{x'}\left(x_n - \underline{x}\right)}_{\text{Difference in Skills}} - \underbrace{\underbrace{\left(x_n - y_n\right)' A \left(x_n - y_n\right)}_{x_n \text{ mismatch at boundary}} - \underbrace{\left(\underline{x} - \underline{y}\right)' A \left(\underline{x} - \underline{y}\right)}_{\underline{x} \text{ mismatch at boundary}}}_{\text{Difference in Mismatch}}$$

- Marginal products and wages reflect:
  1. Skill premium
  2. Mismatch premium

- Differentials depend on assignment through boundary tasks

Roadmap

1. Task assignment model
2. Characterization of solution
   - Assignment
   - Productivity and wages
   - Elasticity of substitution across workers
3. Applications:
   - Directed Automation
   - Automation and unassigned tasks
   - Skill-biased technical change
   - Worker training

Directed Automation

Automation of tasks is a worker replacing technology

- Automation by industrial robots, software
- Offshoring

Automation can be directed by choosing which tasks are automated

- “Low-Skill” tasks are not necessarily automated

It is optimal to automate tasks along the boundaries of occupations

- Replace workers at tasks with high mismatch

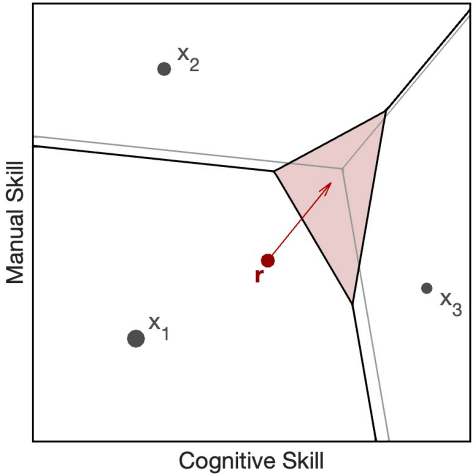
Directed automation: 2 steps

1. Choose robot's mass (p_r) and location in skill space $r = (r^c, r^m)$
   - Engineering the robot is costly: $\Omega(r, p_r)$
2. Assign tasks to workers and robot

$$\max_{\{r, p_r, T\}} \underbrace{F(T, r)}_{\text{Output with Robot}} - \underbrace{\Omega(r, p_r)}_{\text{Automation Cost}} \qquad \text{s.t. } D_n \leq p_n \quad D_r \leq p_r$$

where: $$F(T, r) = \exp\left(\int_{\mathcal{Y}\setminus\mathcal{Y}_r} \ln q(T(y), y)\, dG + \int_{\mathcal{Y}_r} \ln q_R(r, y)\, dG\right)$$

Example: Robot assignment and placement



Assignment:

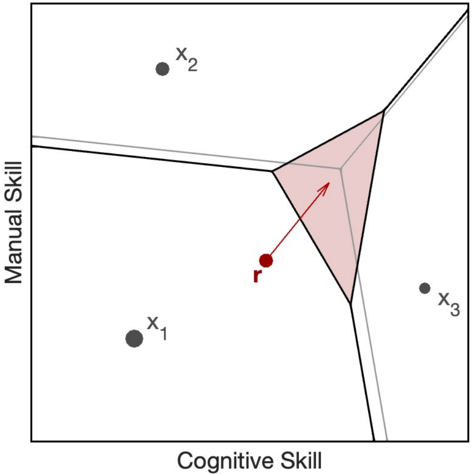
- Tasks with highest mismatch
- Vertices of original assignment

Placement:

- Balance reduction in mismatch with cost of automation

FOC

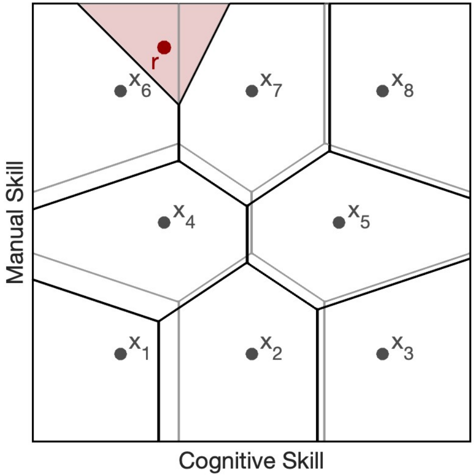
Example: Effects on employment



Task displacement:

- Robot takes tasks from all workers
- Boundaries adjust to maintain employment of x_2, x_3
  - x_2 and x_3 more productive
- Only x_1 is displaced

Example: Cascade Effect



Automation induces a cascade effect

- Workers ordered by marginal products
- Effects on employment not necessarily on workers whose tasks are automated
- Lowest productivity workers unassigned

Effect on wages is ambiguous

- Higher mismatch for workers ($\lambda \downarrow$)
- Higher output ($F(T) \uparrow$)

Roadmap

1. Task assignment model
2. Characterization of solution
   - Assignment
   - Productivity and wages
   - Elasticity of substitution across workers
3. Applications:
   - Directed automation
   - Automation and unassigned tasks
   - Skill-biased technical change
   - Worker training

Unassigned tasks

Previous assumption: Unassigned task $\longrightarrow$ No output $q(\emptyset, y) = 0$

- Consequence: all tasks are assigned

New assumption: Tasks can be left unassigned

- Unassigned tasks are taken out of the aggregate production

$$F(T) = \exp\left(\int_{\mathcal{Y} \setminus \mathcal{Y}_{\emptyset}} \ln q(T(y), y)\, dG \right)$$

- Equivalent to assume $q(\emptyset, y) = 1$

New assignment

Optimal assignment:

$$T(y) = x_n \longleftrightarrow \forall_\ell \ln q(x_n, y) - \lambda_n^\star \geq \ln q(x_\ell, y) - \lambda_\ell^\star \quad \wedge \quad \ln q(x_n, y) - \lambda_n^\star \geq \underline{\lambda}$$

- Where $\underline{\lambda}$ satisfies: $\underline{w} = \underline{\lambda} F(T)$

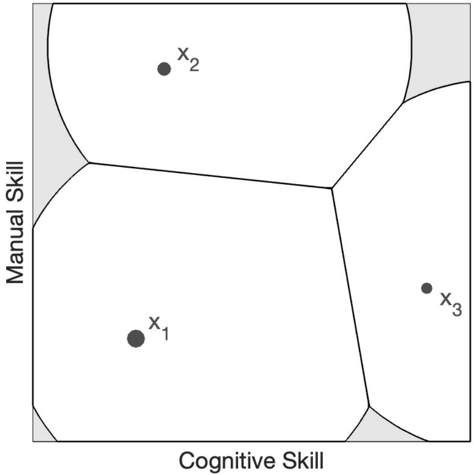
Value of $\underline{w}$ matters for assignment

- If worker is not productive enough task is left unassigned

Workers are left unassigned (even with $\underline{w} = \underline{\lambda} = 0$)

- Necessary condition for assignment: $q(x_n, y) \geq 1$

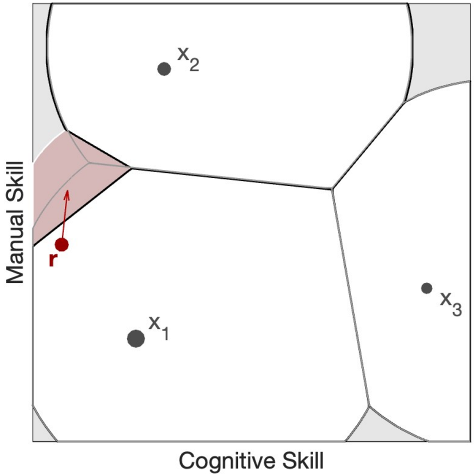
Assignment with unassigned tasks



Unassigned areas in grey

- High mismatch tasks not assigned
- Increasing $\underline{w}$ makes more tasks unprofitable
  - Graph has $\underline{w} = 0$
- Only x_1 is unassigned

Unassigned tasks and automation



Robots are not necessarily labor replacing

- Robot takes over unassigned tasks
- Increase in output:
  - → Increase in wages
  - → Increase in assigned tasks

Roadmap

1. Task assignment model
2. Characterization of solution
   - Assignment
   - Productivity and wages
   - Elasticity of substitution across workers
3. Applications
   - Directed automation
   - Automation and unassigned tasks
   - Skill biased technical change
   - Worker training

Conclusion

Continuous and significant changes in organization of work

- Advances in automation make it possible to replace workers in the workplace
- New technologies biased toward workers with specific skills

I present a framework where occupations react endogenously to these changes

- Multi-dimensional worker's skills and task's skill requirements
- Differences in wages and substitution reflect productivity at boundary tasks

The framework flexibly allows for various applications:

- Automation
- Skill biased technical change

Appendix

Related Literature

- Assignment models:
  - Kantorovich (1942), Koopmans & Beckmann (1957), Sattinger (1975,1984,1993)
- Task models of the labor market:
  - Rosen (1978), Acemoglu & Autor (2011), Autor (2013), among others
- Exploit tools from **Optimal Transport** (Villani, 2009, Galichon, 2016) and **Computational Geometry** (Aurenhammer, 1987, 1991)
  - I solve a Semi-Discrete Optimal Transport problem.

Back

Proof: Existence of a deterministic optimal assignment

The proof follows from applying theorems 5.10 and 5.30 from Villani (2009)

- Before applying the Theorems note that the problem can be relaxed by considering non-deterministic assignments:

$$\pi : \mathcal{X} \times \mathcal{B}\left(\mathcal{Y}\right) \to \mathbb{R}_{+}$$

  π describes assignment as a joint measure over worker/task pairs
- The problem is then:

$$\max_{\pi} \quad \exp\left(\int \ln q\left(x, y\right) \, d\pi\left(x, y\right)\right)$$

$$\text{s.t. } \forall_{n} \quad \int_{\mathcal{Y}} d\pi\left(x_n, y\right) = \int_{\mathcal{Y}_n} dy \leq p_n \qquad \forall_{Y \in \mathcal{Y}} \sum_{n=1}^{N} \pi\left(x_n, Y\right) = \int_{Y} dy$$

- Note that this problem is linear in π

Proof: Existence of a deterministic optimal assignment

- Theorem 5.10 establishes duality for the Planner's problem if $q(x, y) > 0$ and continuous for any pair (x, y)

$$\max_{\pi} \ln F(\pi) = \max_{\pi} \int \ln q(x, y)\, d\pi = \inf_{\substack{(w,v)\in\mathbb{R}^N \times L^1 \\ w_n + v(y) \geq \ln q(x_n, y)}} \sum_{n=1}^{N} \lambda_n(x)\, p_n + \int_{\mathcal{Y}} v(y)\, dy$$

$$= \inf_{\lambda\in\mathbb{R}^N} \sum_{n=1}^{N} \lambda_n p_n + \int_{\mathcal{Y}} \max_{n} \{\ln q(x_n, y) - \lambda_n\}\, dy$$

A solution to the dual problem $(\lambda^\star)$ is guaranteed.

- Galichon (2016) shows an algorithm to find $\lambda^\star$ using the dual problem's FOC

Proof: Existence of a deterministic optimal assignment

- The solution of the dual problem $(\lambda^{\star})$ gives gives the optimal assignment:

$$\forall_y \quad T(y) = \arg\max_{x \in \mathcal{X}} \{\ln q(x,y) - \lambda_x^{\star}\}$$

- Theorem 5.30 gives uniqueness (in law) of the assignment when q is injective given y
  - $T(y)$ is a singleton for almost all y
  - Workers' performance are different at almost all tasks
  - The finiteness of $\mathcal{X}$ simplifies this condition.
- The solution of the dual problem $\lambda^{\star}$ is pinned up to an additive constant. Normalizing to $\min \lambda^{\star} = 0$ follows from the lowest marginal product being zero.
  - If there is an excess of workers $\left(\sum p_n > \int_{\mathcal{Y}} dy\right)$ the level of $\lambda^{\star}$ is also pinned down by $\min \lambda^{\star} = 0$

Back

Marginal product: Two tasks example

- There are two workers (x_n and x_ℓ) and two tasks $\{y_1, y_2\}$
- Total output is given by $F(T) = q_1(x_n)\, q_2(x_\ell)$
  - Worker x_n performs task y_1 and worker x_ℓ performs task y_2

Change assignment by having worker x_n perform both tasks:

- New output is: $F\left(T'\right) = q_1(x_n)\, q_2(x_n) = \frac{q_2(x_n)}{q_2(x_\ell)} F(T)$
- (log) Change in output is:

$$\ln \frac{F\left(T'\right)}{F(T)} = \ln q_2(x_n) - \ln q_2(x_\ell) = \lambda_n - \lambda_k$$

- Output changes by $100\left(\lambda_n - \lambda_k\right)\%$

Back

Elasticity of substitution

back

The (Morishima) elasticity of substitution between workers x_n and x_ℓ is

$$M_{\ell n} = \frac{\partial \ln {}^{D_\ell}/_{D_n}}{\partial \ln \mathrm{MP}_n} = \underbrace{\frac{\mathrm{MP}_n}{D_\ell}\frac{\partial D_\ell}{\partial \mathrm{MP}_n}}_{\mathcal{E}_{\ell n} \text{ Cross Elasticity}} - \underbrace{\frac{\mathrm{MP}_n}{D_n}\frac{\partial D_n}{\partial \mathrm{MP}_n}}_{\mathcal{E}_{nn} \text{ Own Elasticity}}$$

- $\mathcal{E}_{\ell n}$ measures direct substitution between workers

Prop. If q is differentiable with respect to y and $\mathcal{Y}$ is convex then:

i D_n is differentiable wrt $\lambda^\star$ –generalization of Feenstra & Levinsohn (1995)

ii $\mathcal{E}_{\ell n} \geq 0$ with equality if $\mathcal{Y}_l \cap \mathcal{Y}_n = \emptyset$

Elasticity of substitution is determined by boundaries of occupations

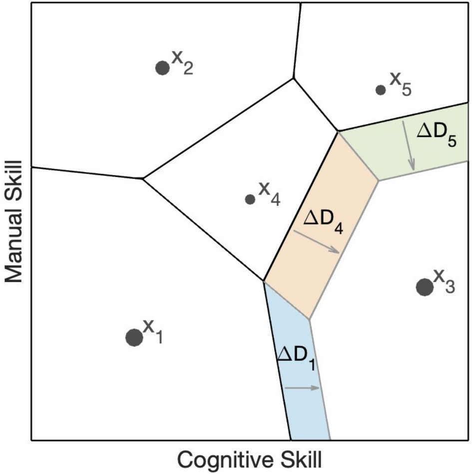
Elasticity of substitution

- Recall that Demand for workers of type n is $D_n = \int_{\mathcal{Y}_n} dy$
  - No close form for D_n in general.
  - Yet, demand is differentiable (Feenstra & Levinsohn, 95)

Key:

- When marginal product changes, the boundaries of the assignment shift in parallel.
- Use the shift in the boundaries to measure the change in demand.

Example: Change in demand - $\lambda_3 \uparrow$



- When $\lambda_3 \uparrow$ it is optimal to assign tasks away of $\mathcal{Y}_3$
- Boundaries move in parallel
- Only the neighbors of x_3 are directly affected.

Proposition: Differentiability of Demand

Let there be at least two skills (i.e. $d \geq 2$) and $\lambda \in \mathbb{R}^N$ be a vector of multipliers.

If q is differentiable wrt y, and $\mathcal{Y}$ convex then D_n is continuously differentiable with respect to λ.

i Change in demand for workers of type n (D_n) when their λ_n changes:

$$\frac{\partial D_n}{\partial \lambda_n} = -\sum_{m \neq n} \frac{\partial D_m}{\partial \lambda_n}$$

- Demand for workers of type n comes from tasks reallocated from other workers.

ii $\frac{\partial D_\ell}{\partial \lambda_n} \geq 0$, with equality if $\mathcal{Y}_l \cap \mathcal{Y}_n = \emptyset$

Proposition: Differentiability of Demand

Back

Further characterization of demand requires a functional form:

$$q(x_n, y) = \exp\left(a_x{}' x_n + -(x_n - y)' A (x_n - y)\right)$$

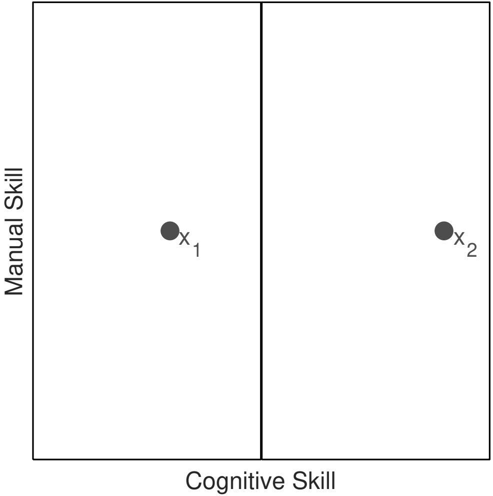
ii If q is as above:

$$\frac{\partial D_\ell}{\partial \lambda_n} = \frac{\int\limits_{\mathcal{Y}_n \cap \mathcal{Y}_m} dG}{2\sqrt{(x_n - x_m)' A' A (x_n - x_m)}}$$

The formula is obtained using Reynold's Transport theorem.

- Substitutability across workers:
  - Increases with exposure: lenght $(\mathcal{Y}_n \cap \mathcal{Y}_m) = \int\limits_{\mathcal{Y}_n \cap \mathcal{Y}_m} dG$
  - Decreases with distance: $(x_n - x_m)' A' A (x_n - x_m)$
- Multidimensional setting allows for more interactions.
  - In one dimension at most two substitutes.

Example: Technical change and skill abundance



- Two workers $\{x_1, x_2\}$:
  - Same manual skill and same mass
- Production technology:
  $$\ln q(x_n, y) = a_x{}' x_n - (x_n - y)' A (x_n - y)$$
- Increase cognitive weight and reduce manual weight in A
- Effect on wages depends on mismatch along boundary:
  - Initial diff. in prod: $\lambda_2 - \lambda_1 = 5.97\%$
  - Final diff. in prod: $\lambda_2 - \lambda_1 = 2.91\%$

Back

Directed automation - Optimality conditions

Back

Necessary conditions for a solution:

$$2F\left(T, r\right) D_r \left(\frac{a_x}{2} - A\left(r - b_r\right)\right) - \Omega_r\left(r, p_r\right) = 0 \qquad [r]$$

$$-\Omega_{p_r}\left(r, p_r\right) + \mu_r = 0 \qquad [p_r]$$

$b_r = \frac{\int_{\mathcal{Y}_r} y dG}{D_r}$: Mean (barycenter) of tasks assigned to robot ($\mathcal{Y}_r$)
μ_r: Robot's marginal product

- The optimal location of the robot is such that it is in the (weighted) center of its region, which minimizes mismatch, adjusted by the weight given to skills in production (a_x) and the cost of those skills (Ω_r)

Directed automation - Optimality conditions

Back

Necessary conditions for a solution:

$$2F\left(T,r\right)D_r\left(\frac{a_x}{2}-A\left(r-b_r\right)\right)-\Omega_r\left(r,p_r\right)=0 \qquad [r]$$

$$-\Omega_{p_r}\left(r,p_r\right)+\mu_r=0 \qquad [p_r]$$

$b_r=\frac{\int_{\mathcal{Y}_r} ydG}{D_r}$: Mean (barycenter) of tasks assigned to robot ($\mathcal{Y}_r$)
μ_r: Robot's marginal product

- Optimal location of the robot is such that it is in the (weighted) center of its region, which minimizes mismatch, adjusted by the weight given to skills in production (a_x) and the cost of those skills (Ω_r)

Directed automation - Optimality conditions

Back

Necessary conditions for a solution:

$$2F(T,r)\,D_r\left(\frac{a_x}{2} - A(r - b_r)\right) - \Omega_r(r, p_r) = 0 \qquad [r]$$

$$-\Omega_{p_r}(r, p_r) + \mu_r = 0 \qquad [p_r]$$

$b_r = \frac{\int_{\mathcal{Y}_r} y dG}{D_r}$: Mean (barycenter) of tasks assigned to robot ($\mathcal{Y}_r$)
μ_r: Robot's marginal product

- Optimal location of the robot is such that it is in the (weighted) center of its region, which minimizes mismatch, adjusted by the weight given to skills in production (a_x) and the cost of those skills (Ω_r)

Directed automation - Optimality conditions

Back

Necessary conditions for a solution:

$$2F(T,r)\,D_r\left(\frac{a_x}{2} - A(r-b_r)\right) - \Omega_r(r,p_r) = 0 \qquad [r]$$

$$-\Omega_{p_r}(r,p_r) + \mu_r = 0 \qquad [p_r]$$

$b_r = \frac{\int_{\mathcal{Y}_r} y dG}{D_r}$: Mean (barycenter) of tasks assigned to robot ($\mathcal{Y}_r$)
μ_r: Robot's marginal product

- Robot's mass is increased (higher p_r) until marginal cost of (Ω_{p_r}) equals robot's marginal product (μ_r)

Skill Biased Technical Change

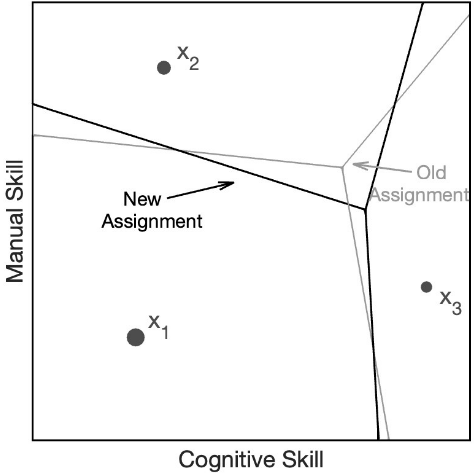
Skill biased technical change

$$q\left(x_n, y\right) = \exp\left({a_x}' x_n - \underbrace{\left(x_n - y\right)' A \left(x_n - y\right)}_{\text{Worker/Task Mismatch}} \right) \qquad \text{where: } A = \begin{bmatrix} \alpha & 0 \\ 0 & 1-\alpha \end{bmatrix}$$

- Technology changes how skill mismatch affects production: A
  - α controls relative importance of cognitive mismatch in production
  - Use of machinery $\longrightarrow$ Reduce importance of manual mismatch ($\alpha \uparrow$)
  - Use of computers $\longrightarrow$ Increase importance of cognitive mismatch ($\alpha \uparrow$)

- Two types of effects:
  - Direct effect through productivity of workers at boundary tasks
  - Reassignment effect through changes in the tasks performed by workers

Direct and reassignment effects of $\alpha \uparrow$



Direct effect:

- Differences in wages depend on differences on cognitive skills
  - Technical change does not necessarily benefit abundance of skill Example
- x_1 and x_2 become more substitutable

Reassignment effect:

- Minimize cognitive mismatch

Direction of skill biased technical change

- The relative importance of skills is governed by matrix A
  - In what follows I impose additional structure on A:

$$A = \begin{bmatrix} \alpha & 0 \\ 0 & 1-\alpha \end{bmatrix}$$

  - Higher α makes cognitive match more important for production
- Changing α enhances the workforce by putting more weight on skills for which the workforce is better suited

Choose: skill weight (α) and assignment (π) to maximize output

$$\max_{\{\alpha,\pi\}} \quad \underbrace{F(\pi,\alpha)}_{\text{Output with Innovation}} \quad - \quad \underbrace{h(\alpha)}_{\text{Innovation Cost}}$$

Technology is chosen to minimize mismatch

Back

The optimal α satisfies:

$$F(\pi, r)(M_m - M_c) - h_\alpha(\alpha) \geq 0 \qquad [\alpha]$$

Where M_s is total mismatch in skill s: $M_s = \sum_{n=1}^{N} \int_{\mathcal{Y}_n} (x_{n,s} - y_s)^2 \, dy$

- Total mismatch depends on assignment and distribution of tasks and workers
- If there is more mismatch in the manual dimension ($M_m > M_c$):
  - The workforce is biased (in equilibrium) towards cognitive skills
  - Technical change is directed towards cognitive skills ($\alpha \uparrow$)
  - Technology reinforces bias by weighting skills with better match

Higher weight on cognitive skills

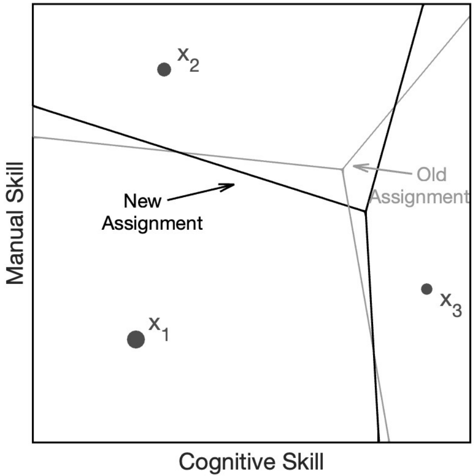
Information technology and computer use change the way tasks are completed

- Higher weight on cognitive skills

Effect on workers:

- Cognitive differences across workers have larger effects on output
  - Workers with high cognitive skills benefit
  - Wage differences across workers with different cognitive skills increase
- Differences in manual skills are less important
  - Wage differentials between workers with different manual skills shrink

Example: Higher weight on cognitive skills



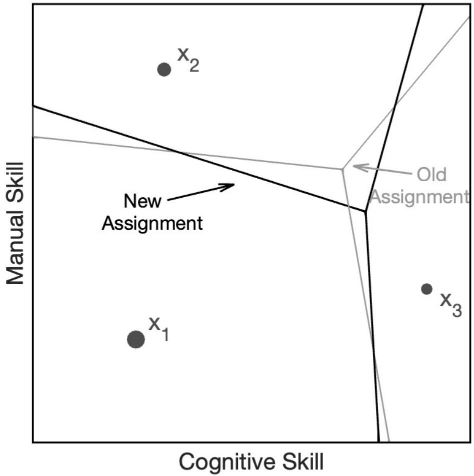
Assignment biased towards minimizing cognitive mismatch

- Partition is more "vertical"
- Tasks using high cognitive skills reallocated to x_3
- Wage of x_3 workers $\uparrow$ 12.3%
- Wage premium of cognitive-intensive worker $\uparrow$

$$50\% \longrightarrow 70\%$$

Back

Example: Higher weight on cognitive skills



Differences between x_1 and x_2 are less important

- Similar cognitive skills $\rightarrow$ x_1 and x_2 more substitutable
- Wage of x_2 workers $\downarrow$ 6.8%
- Wage of x_1 workers unchanged

Back

Optimal Worker Training

Optimal worker training

The automation framework applies to the question of worker training:

- What skills ($\tilde{x}$) should be given to a worker?
- Choose new skills $\tilde{x}$ for worker x_n, instead of robot skills (r)

Higher gains from training workers with higher mismatch

- Change skills to reduce mismatch, more skills are not always better

Changing skills of one worker changes assignment of other workers

- Mismatch can increase for other workers $\longrightarrow$ ambiguous effect on wages

Optimal worker training

The problem is to choose skills for the worker ($\tilde{x}$) and a new assignment (π):

$$\max_{\{\tilde{x},T\}} F\left(T,\tilde{x}\right) - h\left(\tilde{x}|x_{n,}p_n\right)$$

- $h\left(\tilde{x}|x_{n,}p_n\right)$ is the cost of changing skills x_n to $\tilde{x}$ for p_n workers
- I am assuming that all (p_n) workers of type x_n are trained
  - The problem is the same if the workers are into M groups
  - An additional cost for specialization must be added, increasing in M

The optimality condition is:

$$0 = 2F\left(T,\tilde{x}\right) D_n \left(\frac{a}{2} - A\tilde{x} + Ab_n\right) - \frac{\partial h\left(\tilde{x}|x_{n,}p_n\right)}{\partial \tilde{x}}$$

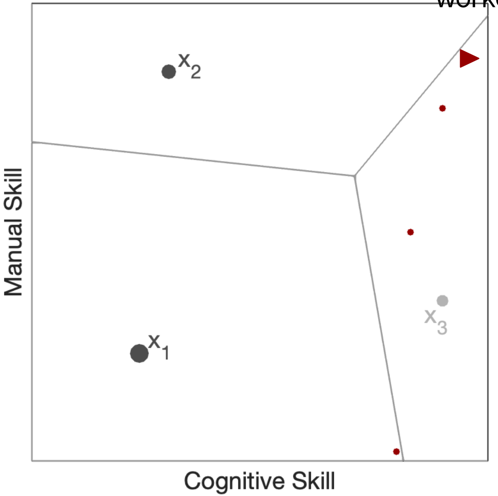
Back

Increase in specialization

Increase in specialization

- Changes in college education tend towards higher specialization:
  - Increase in post-graduate education
  - Increase in the number of majors
- Specialized workers tend to earn higher wages
  - Specialized workers perform a smaller set of tasks and have lower mismatch
  - Specialization in only one skill can bring costs
- As some workers specialize the assignment changes for all workers
  - Occupation boundaries tasks respond to new distribution of workers

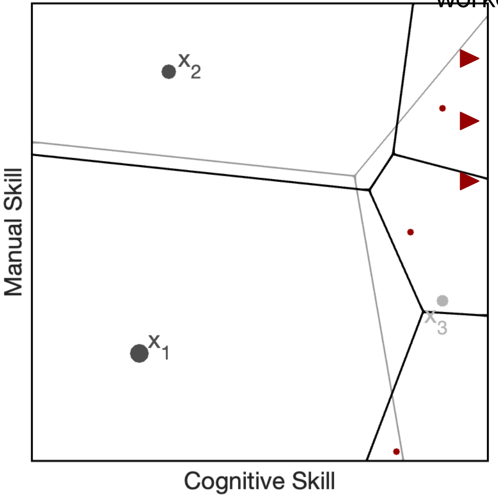
Example: Specialization increases wages



Break up the mass of cognitive-intensive workers (x_3)

- Equal mass among new workers

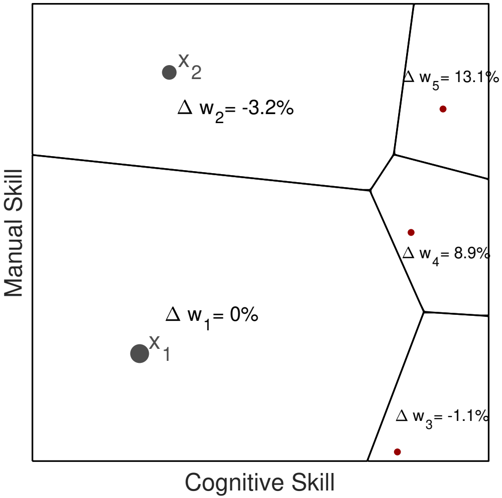
Example: Specialization increases wages



Break up the mass of cognitive-intensive workers (x_3)

- Equal mass among new workers
- Boundaries of occupations change
- Higher output: $\uparrow$ 0.6%
  - Mismatch $\downarrow$ for x_3
  - Mismatch $\uparrow$ for x_1, x_2

Example: Specialization increases wages



- Av. wage of specialized workers ↑ 3.4%

Gain differs across workers

- Top worker gains the most: Mismatch ↓ + Skills ↑
- Bottom worker looses: Mismatch ↓ + Skills ↓

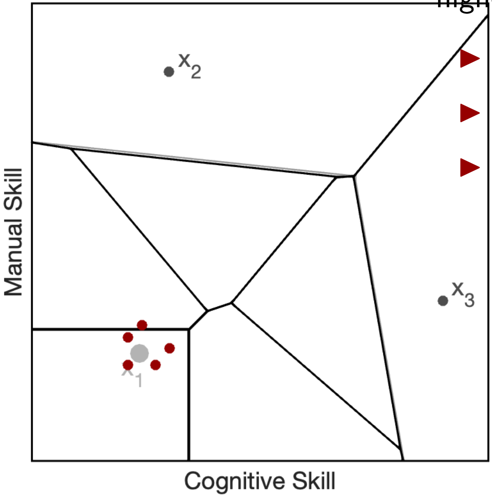
Wage bill ↑ 0.7%

Labor share stable 47.5% to 47.6%

Automation and specialization

- Specialization can hurt workers when tasks are automated
- Automation can be concentrated in tasks assigned to a specialized worker
  - The worker is displaced or reassigned
  - Mismatch increases for the worker after reassignment
- Consider the specialization example from before, with automation

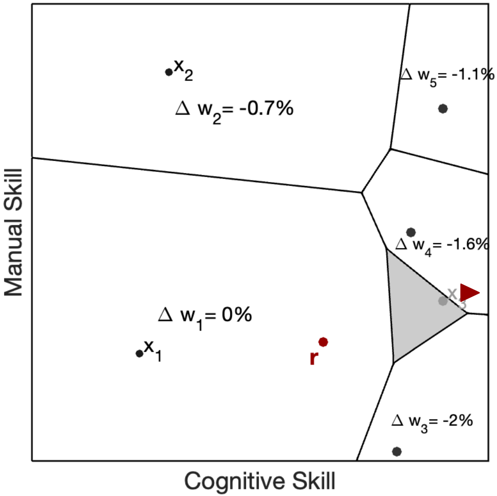
Example: Automation and specialization



Robot is optimally assigned to tasks with highest mismatch

- Area was previously covered by x_3
- Large mismatch with x_1
- Takes over tasks of specialized workers

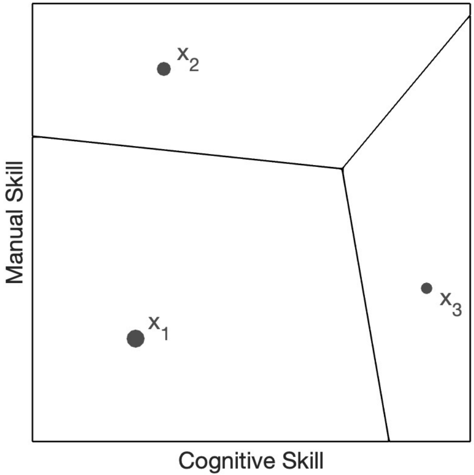
Example: Automation and specialization



- Wages go down more than in previous example
  - Old x_3 wage $\downarrow$ 1.2%
  - Specialized workers's wage $\downarrow$ 1.9% and 1.3%
  - For bottom worker that adds to initial wage decrease
- Output $\uparrow$ 0.4%

Changes in Worker Supply

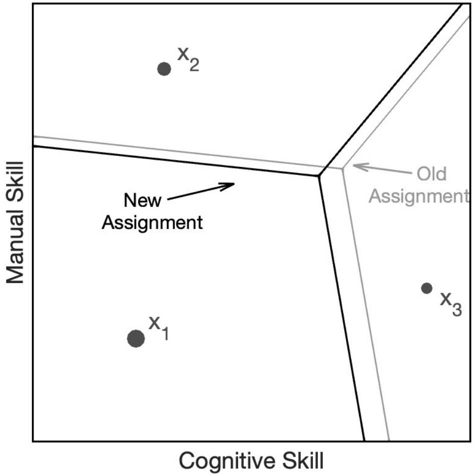
Example: Increase in cognitive intensive workers



More cognitive-intensive workers and less low-skilled workers

- The supply of worker x_3 increases: $p_3 \uparrow$ from 20% to 25%
- The supply of worker x_1 decreases: $p_1 \downarrow$ from 50% to 45%

Example: Increase in cognitive intensive workers



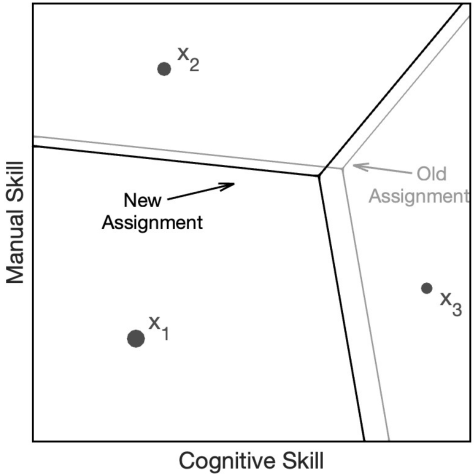
The assignment changes by adjusting wages

- $w_3 \downarrow$ (6%) x_3 takes over more tasks, poorer matches
- w_1 is unchanged
- $w_2 \downarrow$ (2.5%) to clear market

Higher output: $\uparrow$ 1%

- x_3 higher productivity (a_x) compensates higher mismatch

Example: Increase in cognitive intensive workers



Wage bill goes down (0.6%)

- Despite change in workforce $x_1 \to x_3$

Labor share goes down:

$$47.5\% \to 46.8\%$$

- Lower wages and higher output