# Replication Material for "Exam Reward Structure, Gender Performance Gaps, and Labor Market Outcomes"

Bruna Borges[1]

Fernanda Estevan[2]

Louis-Philippe Morin[3]

February 18, 2025

[1] Sao Paulo School of Economics - FGV, Rua Dr. Plinio Barreto, 365, Sao Paulo, SP, Brazil; email: bruna.borges@fgv.br.

[2] Sao Paulo School of Economics - FGV, Rua Dr. Plinio Barreto, 365, Sao Paulo, SP, Brazil; email: fernanda.estevan@fgv.br.

[3] Department of Economics, University of Ottawa, 120 University, Ottawa ON K1N 6N5, Canada; email: lmorin@uottawa.ca.

# 1. Overview

The replication package described here contains all the code used to generate results for “Exam Reward Structure, Gender Performance Gaps, and Labor Market Outcomes” by Borges, Estevan, and Morin published in the *Journal of Labor Economics.* The data source combines data from UNICAMP’s admission office *(Comissão Permanente para os Vestibulares, COMVEST*) with employer-employee data from *Relação Anual de Informações Sociais* (RAIS). These data files are not publicly available. Researchers interested in accessing the data can do so by following the procedure outlined in Section 2. Those who receive access can reproduce the results presented in the paper using the code included in the replication package. This folder contains all necessary code (do-files) to construct the datasets and reproduce all results (tables and figures) in the main manuscript and in the Online Appendix. All codes can be run with Stata 17.

# 2. Data availability and provenance

This paper relies on proprietary data, and the replication files include all code files necessary to reproduce the paper’s results. Below is a detailed description of how the data was obtained, allowing others to reproduce the dataset.

## 2.1 Statement about Rights

The authors of this manuscript have the proper authorization and rights to access and utilize the data included in this study.

## 2.2 Summary of Availability

The data cannot be made publicly available.

## 2.3 Details on Data Source

a. UNICAMP administrative dataset

The main datasets used in our analysis are administrative data provided by UNICAMP’s admission office *(Comissão Permanente para os Vestibulares, COMVEST*). The data is proprietary and cannot be made publicly available. Requests for UNICAMP data should be directed to pesquisa@comvest.unicamp.br. Once a request is approved, applicants must sign a confidentiality agreement.

Before creating the main dataset, we combined the original UNICAMP data into three .dta files: 1) work_data_P1.dta, 2) work_data_P2.dta, 3) work_data_SE.dta. These datasets are generated using the scripts “Phase1.do”, “Phase_2.do”, and “Socioeconomic.do”.

The script “Phase1.do” performs several tasks to process the Phase 1 outcome data. First, it merges the original files (e.g., V2002_Fase1.txt) into a single .dta file (work_data_P1.dta). Then, it renames the variables for

consistency and addresses issues related to missing observations. The original data includes applicant identification numbers, a binary indicator for exam participation, the exam year, individual section scores from Phase 1, the final Phase 1 score, and the applicant's ENEM score.

The script "Phase2.do" processes the Phase 2 outcome data. It first merges the original files (e.g., V2003_Fase2.txt) into a single .dta file (work_data_P2.dta). Then, it renames the variables for consistency, resolves issues with missing observations, and creates an exam-year variable. The original dataset includes applicant identification numbers, scores for each of the twelve questions across eight subjects (totaling 96 variables), and grades from aptitude tests required for specific fields such as Music, Arts, and Dentistry.

The script "Socioeconomic.do" processes applicants' socioeconomic data. It merges two datasets (e.g., V2004_QSEPR.txt and V2004_QSETT.txt) into a single .dta file (work_data_SE.dta). Then, it renames the variables for consistency, creates an exam-year variable, and recodes certain programs to ensure uniform identification across different years.

The original VYEAR_QSEPR.txt and VYEAR_QSETT.txt datasets include various socioeconomic information about applicants, such as gender, age, and career choice code. They also contain information on whether the applicant advanced to the second stage of the admission exam, enrolled at UNICAMP, and, if enrolled, their chosen major. Additionally, the datasets record the type of secondary school attended and whether the applicant participated in a preparatory course for the admissions test. The files also include the applicant number, enabling us to merge information with Phase 1 and Phase 2 exam data.

The script "Main_Dataset.do" constructs the main dataset used in our analysis by merging multiple data sources and applying necessary transformations. In the first step, it merges the socioeconomic data, Phase 1 scores, and Phase 2 scores while performing data cleaning and generating new variables. The second step involves creating the final score variables. Next, it applies sample restrictions as outlined in the Data Section of our paper, such as focusing on Phase 1 survivors, excluding trainees, etc. Following this, the script reshapes the data from a wide to a long format, ensuring that each observation corresponds to a subject per applicant in Phase 2, resulting in eight rows per candidate. Finally, it normalizes the variables, preparing them for the empirical analysis.

We have also included in the replication package a master script for the UNICAMP dataset "Data_UNICAMP.do" that runs all the do-files regarding the UNICAMP administrative data that we described in this subsection.

b. Relação Anual de Informações Sociais (RAIS)

The dataset used is a matched employer-employee database covering the universe of the Brazilian formal labor market. It is compiled annually by the

Brazilian government and includes records of all formal-sector employers and employees. The data provide detailed information on employment contracts, such as start dates, wages, occupations, hours worked, and firm characteristics like industry, workforce size, and location. While an anonymized version is publicly available through the Ministry of Labor and Employment (MTE), our analysis required the identified dataset containing workers' names, individual taxpayer numbers (CPF), birth dates, etc. The merge was performed by the Computer Science Institute at UNICAMP using workers' names and birth dates. They provided us the labor market information with the same COMVEST applicant identification numbers.[4] This enabled us to merge it with administrative records from UNICAMP. Access to the identified data is granted through agreements between the MTE and Brazilian higher education institutions, with access protocols varying by institution and evolving over time.

The script "Data_RAIS.do" creates the labor market data used in our analysis.

# 3. Computational requirements

All codes can be run with Stata 17.

We have installed the following Stata packages:

- estout from http://fmwww.bc.edu/RePEc/bocode/e
- ftools from http://fmwww.bc.edu/RePEc/bocode/f
- gtools from http://fmwww.bc.edu/RePEc/bocode/g
- mdesc from http://fmwww.bc.edu/repec/bocode/m
- oaxaca from http://fmwww.bc.edu/repec/bocode/o
- reghdfe from http://fmwww.bc.edu/RePEc/bocode/r
- rif from http://fmwww.bc.edu/RePEc/bocode/r
- _gwtmean from http://fmwww.bc.edu/repec/bocode/_

# 4. Analysis: Description of programs

a. Main Text

The do-file "Main_Text_Tables.do" generates all tables from the main text, while the do-file "Main_Text_Figures.do" reproduces all figures from the main manuscript.

As we only present Figure 1 in the main manuscript, "Main_Text_Figures.do" directly creates this figure. First, it creates the necessary variables to run the regressions and the adjusted score for simulations. Second, it performs

[4] The procedure for data merging and confidentiality assurance was approved by the UNICAMP Research Ethics Committee under process number 25308719.4.0000.8142.

simulations of admission rates without gender gaps in priority subjects (lower bound). Last, it creates graphs to visualize the female and men actual and adjusted admission rates for female and male applicants and exports them to pdf files.

The script “Main_Text_Tables.do” is a master script that runs all tables (each in a separate do-file) in the correct order:

- Table1.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions using normalized Phase 1 score as the dependent variable, and exports the table to a .tex file.

- Table2.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- Table3.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions for the heterogeneity impact of the priority subjects across subjects using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- Table4.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions for the heterogeneity impact of the priority subjects across academic ability using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- Table5.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions presented in Table 5, and exports the table to a .tex file. The code runs the regressions for the dependent variables used to analyze the response pattern: Number of omissions, number of attempted questions with wrong answers (zero scores), and number of zeros (zero scores and omissions).

- Table6.do: creates the necessary variables, collapses the data set at the individual level, runs the regressions where the response variable is the coefficient of variation for each applicant over all eight P2 subjects and gender dummy and ENEM scores are covariates, and exports the table to a .tex file.

- Table 7.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table 7, and exports the table to a .tex file. First, the code

runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- Table8.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table 8, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure by gender. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) by gender and export the results from the bootstrap and wage regressions to .tex file.

b. Online Appendix

The do-file "OnlineAppendix_Tables.do" generates all tables from the Online Appendix, while the do-file "OnlineAppendix_Figures.do" reproduces all figures from the Online Appendix.

The script "OnlineAppendix_Tables.do" is a master script that runs all tables (each in a separate do-file) in the correct order:

- TableO1.do: creates the necessary variables, collapses the data set at the individual level, runs the regressions where the response variable is the number of priorities (first major choice and all major choices), and gender dummy, ENEM scores, and first major fixed effects are covariates, and exports the table to a .tex file.

- TableO2.do: creates the necessary variables, calculates descriptive statistics related to each major, and exports the table to a .xlsx file.

- TableO3.do: creates the necessary variables, collapses the data set at the individual level, runs the regressions where the response variables are Phase 2 scores and the key explanatory variables Phase 1 scores, and exports the table to a .tex file.

- TableO4.do: creates the necessary variables, collapses the data set at the individual level, calculates descriptive statistics related to each applicant, and exports the table to a .tex file.

- TableO5.do: creates the necessary variables, calculates descriptive statistics summarizing the score in each subject for each applicant, and exports the table to a .tex file.

- TableO6.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions using normalized Phase 1 score as the dependent variable, and exports the table to a .tex file. This do-file uses the dataset that covers all applicants. We use a script similar to "Main_Dataset.do," but do not exclude those candidates who did not qualify for the second phase.

- TableO7.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, restricts the sample to applicants that did not qualify for Phase 2, runs the main regressions by quintiles using normalized Phase 1 score as the dependent variable, and exports the table to a .tex file. This do-file uses the dataset that covers all applicants. We use a script similar to "Main_Dataset.do", but do not exclude those candidates who did not qualify for the second phase. We also run the main regression (Table 2) to include in the table for comparison purposes.

- TableO8.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions for the heterogeneity impact of the priority subjects across exam day, and exports the table to a .tex file.

- TableO9.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, excludes Medicine applicants (UNICAMP and FAMERP), runs the main regressions using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO10.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, excludes Medicine applicants (UNICAMP and FAMERP), runs the regressions for the heterogeneity impact of the priority subjects across subjects, and exports the table to a .tex file.

- TableO11.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis and Portuguese (using the essay as its Phase 1 score), runs the main regressions using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO12.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions using raw Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO13.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions using Phase 2 score normalized by year, subject and gender as the dependent variable, and exports the table to a .tex file.

- TableO14.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including additional interactions with Phase 1 scores, using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO15.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions using an alternative dependent variable (normalized Phase 2 - Phase 1 Scores), and exports the table to a .tex file.

- TableO16.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions including 'All P1 Scores X Subject FE', using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO17.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including interactions with the priority subjects related to the second or third choices, using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO18.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including interactions with a difficult dummy (equal to one if the average performance on this question is below the median of question average scores for that subject-year) and using question's raw scores (ranging from 0 to 5 points) as the dependent variable, and exports the table to a .tex file.

- TableO19.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including interactions with a difficult dummy (equal to one if the average performance on this question is below the median of question average scores for that subject, year and gender) and using

question's raw scores (ranging from 0 to 5 points) as the dependent variable, and exports the table to a .tex file.

- TableO20.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including interactions with a very difficult dummy (bottom 25% of question average scores for that subject-year) and using question's raw scores (ranging from 0 to 5 points) as the dependent variable, and exports the table to a .tex file.

- TableO21.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions, including interactions with a most difficult dummy (equal to one if the question has the lowest question average score for that subject-year) and using question's raw scores (ranging from 0 to 5 points) as the dependent variable, and exports the table to a .tex file.

- TableO22.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the regressions using the average raw score in early questions (1 to 4) minus the average raw score in the late questions (9 to 12) for each subject in Phase 2 as the dependent variable, and exports the table to a .tex file.

- TableO23.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, splits the applicants into two subsamples: 'First time UNICAMP' and 'Not first time', runs the main regressions for each subsample using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO24.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, splits the applicants into two subsamples: 'Preparatory course' and 'No preparatory course', runs the main regressions for each subsample using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO25.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, splits the applicants into two subsamples according to the school location: 'Campinas metropolitan region' and 'Other cities', runs the main regressions for each subsample using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO26.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, splits the applicants

into two subsamples related to their secondary schools: ‘Public school’ and ‘Private school’, runs the main regressions for each subsample using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO27.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, splits the applicants into two subsamples: ‘At least one parent has higher education degree’ and ‘No parent has higher education degree’, runs the main regressions for each subsample using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO28.do: creates the necessary variables, runs regressions by subject and gender using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO29.do: creates the necessary variables, restricts the sample to female applicants, runs regressions by subject using normalized Phase 2 score as the dependent variable and controlling for all Phase 1 scores separately, and exports the table to a .tex file.

- TableO30.do: creates the necessary variables, restricts the sample to male applicants, runs regressions by subject using normalized Phase 2 score as the dependent variable and controlling for all Phase 1 scores separately, and exports the table to a .tex file.

- TableO31.do: creates the necessary variables, runs regressions by subject using normalized Phase 2 score as the dependent variable and controlling for all Phase 1 scores separately, interacting coefficients with a gender dummy to compute coefficients and standard errors for gender gaps, and exports the table to a .tex file.

- TableO32.do: creates the necessary variables, excludes Medicine applicants (UNICAMP and FAMERP), runs regressions by subject and gender using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO33.do: creates the necessary variables, runs regressions by subject and gender using the number of attempted questions as the dependent variable, and exports the table to a .tex file.

- TableO34.do: creates the necessary variables, excludes Medicine applicants (UNICAMP and FAMERP), runs regressions by subject and gender using the number of attempted questions as the dependent variable, and exports the table to a .tex file.

- TableO35.do: creates the necessary variables, estimates the IRT model and stores the predicted scores in a dataset, restricts the sample to the subjects we focus on in our main analysis, restricts the sample to years 2001-2002, replaces omissions for predicted IRT score and calculates new normalized Phase 2 score, runs the main regressions using the modified normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO36.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, restricts the sample to years 2001-2002, runs the main regressions using normalized Phase 2 score as the dependent variable, and exports the table to a .tex file.

- TableO37.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, restricts the sample to years 2001-2002, runs the regressions for the heterogeneity impact of the priority subjects across subjects where the response variables are the number of omitted questions, and exports the table to a .tex file.

- TableO38.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.38, and exports the table to a .tex file. First, the code runs the regression presented in column (7) from Online Appendix O.16, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO39.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.39, and exports the table to a .tex file. First, the code runs regressions for a simpler specification, in which P2 scores are the response variables, P1 scores are the control variables, and the residuals are obtained. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO40.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.40, and exports the table to a .tex file. First, the code runs the main regression, excluding the 'Female × Priority' interaction and the 'ENEM × Priority' and 'Phase 1 scores × Priority' polynomials, then obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO41.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.41, and exports the table to a .tex file. First, the code runs the main regression and obtains the residuals. Next, we add the 'Priority' coefficients for priority subjects for male students and the 'Priority' and 'Female × Priority' coefficients for female students. We calculate the difference in average residuals between priority and non-priority subjects ('relative priority performance'). We calculate the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO42.do: creates the necessary variables, runs the exercise displayed in Table O.42, and exports the table to a .tex file. First, the script runs regressions by subject and gender using normalized Phase 2 score as the dependent variable and obtains the residuals for each regression. The script calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO43.do: creates the necessary variables, runs the exercise displayed in Table O.43, and exports the table to a .tex file. First, the script runs regressions by subject and gender using normalized Phase 2 score as the dependent variable and controlling for all Phase 1 scores separately. Then, it obtains the residuals for each regression. The script calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages

between 7 and 12 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO44.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.44, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 6 and 14 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO45.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.46, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the wage regressions (log annual wages between 6 and 9 years after the admission exam) and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO46.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.46, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the regressions where the response variable is formal labor market participation between 7 and 12 years after the admission exam and exports the results from the bootstrap and wage regressions to the .tex file.

- TableO47.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.47 and detailed in the section 'O.3 Trimming Exercise', and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority

performance'). Next, it checks the impact on selection and implements the trimming exercise. It randomly and incrementally drops applicants with priority performance above the median until the point estimate for relative priority performance is close to zero (magnitude smaller than 0.001). After dropping the observations, it estimates and stores the coefficient estimates from the wage regressions (log annual wages between 7 and 12 years after the admission exam) with the trimmed samples. By iterating 500 times, it obtains a distribution of the estimates of interest approximating the range of potential estimates.

- TableO48.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the Blinder-Oaxaca Decomposition displayed in Table O.48, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it decomposes wage regressions using Blinder-Oaxaca Decomposition and exports its results.

- TableO49.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the quantile regressions displayed in Table O.49, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs recentered influence function (RIF) regressions to estimate quantile wage regressions for the quantiles 10, 25, 50, 75, and 90 and export the results.

- TableO50.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise displayed in Table O.50, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the regressions where the response variables are formal labor market participation for each year between 6 and 14 years after the admission exam. It also exports bootstrap and wage regressions results to the .tex file.

- TableO51.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the exercise

displayed in Table O.51, and exports the table to a .tex file. First, the code runs the main regressions, excluding the 'Female × Priority' interaction, and obtains the residuals. It calculates the difference in average residuals between priority and non-priority subjects ('relative priority performance'). Next, it calculates the standard errors using the cluster-bootstrap procedure. Finally, it runs the regressions where the response variables are dummy variables indicating if the applicant earned a higher education degree between 6 and 14 years after the admission, between 7 and 12 years after the admission, or at any moment. It also exports the bootstrap and wage regressions results to the .tex file

The script "OnlineAppendix _Figures.do" is a master script that runs all tables (each in a separate do-file) in the correct order:

- FigureO1.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, collapses the average outcomes (raw score and binary variables for perfect score, omission, and zero score) at the question's order of appearance in the exam, plots the graphs and exports them to pdf files.

- FigureO2.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, reshapes data to long format, collapses the data at the question's order of appearance in the exam to obtain the share of difficulty questions per order, plots the graphs and exports them to pdf files.

- FigureO3.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, plots the binscatters by subject, correlating Phase 2 and Phase 1 scores, and exports them to pdf files.

- FigureO4.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs main regressions using questions' raw score by order of appearance in the exam, stores the coefficients and confidence intervals, plots the estimates by order and exports them to pdf files.

- FigureO5.do: creates the necessary variables, estimates the IRT model and store the predicted scores in a dataset, restricts the sample to the subjects we focus on in our main analysis, restricts the sample to years 2001-2002, calculates the IRT residual (score - predicted score) and its standard deviation, plots the kernel density of the standard deviation of IRT residuals by gender and priority status, and exports them to pdf files.

- FigureO6.do: creates the necessary variables to run the regressions and the adjusted score for simulations. Second, it performs simulations of admission rates without gender gaps in priority subjects (upper bound). Last, it creates graphs to visualize the female and men actual and adjusted admission rates for female and male applicants and exports them to pdf files.

- FigureO7.do: creates the necessary variables, restricts the sample to the subjects we focus on in our main analysis, runs the main regressions excluding the ‘Female × Priority’ interaction, and obtains the residuals. The script calculates the difference in average residuals between priority and non-priority subjects (‘relative priority performance’). Last, plots the kernel density of the ‘relative priority performance’ by gender