Yuval Margolin, 31944434
Daniel Maor, 211903752

Causal Inference: Project Proposal

1. Our Causal Question

In our project, we will attempt to answer the following question:

"Does working remotely (versus fully in-office) causally affect total compensation for software developers In the United States?"

In particular, we will examine U.S.-based professional developers in recent years. The treatment is "fully remote work" (with no in-office requirement) compared to the control, which is "fully in-person work". The outcome is the developer's total annual compensation (in Dollars). We aim to estimate the causal effect of remote work on salary, holding other factors constant. This question is motivated by the rise of remote work in tech and debates about whether remote employees earn more, less, or the same as on-site peers.

Causal Direction and Timing

We assume that remote work status affects compensation, rather than the other way around. In most professional settings, work arrangements are agreed upon before salaries are finalized, and pay is often set based on the role's structure, including whether it is remote or in-person. While higher-paid or more senior developers may have more flexibility to choose remote roles, we address this by controlling for experience, job level, and related covariates. Given that the survey captures current work status and salary together, we treat remote status as preceding or coinciding with salary determination.

2. Existing Knowledge and Literature

Recent research has begun to examine whether remote work has a causal impact on employee compensation in the U.S. labor market. In particular, two studies closely related to our question examined the relationship between working remotely (versus in-office) and wages. Both studies focus on the pandemic-driven shift to remote work, which provides a natural context to observe how compensation patterns changed when many jobs moved out of the office.

A study by BLS compared wages of remote and on-site workers, using American Community Survey microdata [2]. To examine the relationship between working remote and wages, the authors used several strategies: (1) estimate a linear model by OLS with control variables to address selection on a rich set of observables, (2) estimate bounds on the OLS estimates based on Oster's method that relates selection on observables to selection on unobservables, and (3) estimate relationships across heterogenous groups of workers where selection may be more or less prevalent[1]. They found that remote workers earned about 13% higher wages on average than comparable in-office workers during 2020–2021, with larger premiums in some occupations (e.g., 20% in sales, 17% in management). The results remained robust after controlling for demographics, job characteristics, and potential confounders, suggesting that remote work itself may boost wages.

[1] The strategies description was taken directly from the BLS paper to maintain accuracy.

Yuval Margolin, 31944434
Daniel Maor, 211903752

Additionally, a 2022 study by Barrero et al examined whether firms strategically use remote work policies as a tool to moderate wage growth pressures. Rather than comparing individual remote versus in-office workers' compensation directly, the authors surveyed business executives to investigate the employer side of the wage-setting process. Using novel data from the Survey of Business Uncertainty, they asked firms whether they expanded remote work opportunities specifically to moderate wage-growth pressures and quantified the magnitude of this effect. The study found that 38% of firms expanded remote work opportunities over the previous year to moderate wage growth, with an estimated cumulative wage-growth moderation of 2.0 percentage points over two years. While this approach does not directly answer whether remote work causally affects individual compensation levels, it provides complementary evidence from the employer perspective, suggesting that firms view remote work as having sufficient amenity value to workers that it can serve as a substitute for wage increases. This employer-side evidence helps contextualize findings like those from the BLS study by illuminating one potential mechanism through which remote work and wages may be related.

3. Our Data: Stack Overflow Developer Survey (2022–2024)

In this research, we will use observational survey data from the Stack Overflow Annual Developer Surveys, 2022–2024. These surveys collect responses from tens of thousands of developers worldwide each year(in total, 228000). Crucially, the data include each respondent's work arrangement (remote/hybrid/in-office), total compensation, work experience, position, and various demographic and job-related characteristics. When analyzing the data, we restricted the sample to U.S.-based professional developers (to avoid cross-country differences in pay scales and purchasing power). We define "remote" strictly as those who work fully remote (100% from home or off-site) versus "in-office" as those fully on-site (0% remote). Developers who work in a hybrid arrangement will be excluded or possibly analyzed separately to ensure a clear binary treatment definition. The outcome variable is total annual compensation in USD, which the survey collects as a numeric entry for salary (including bonuses, perks, etc.). In total, we are left with 9985 Remote workers and 1699 In-person workers (Before trimming).

The key covariates we use as potential confounders include years of coding/professional experience, highest education level, job role or developer type, industry type, and company size, among others. These are plausible confounders because they affect both the likelihood of working remotely and salary levels. For example, industry, company size, and role influence the availability of remote positions and are also strong predictors of pay. Likewise, experience and education shape both career opportunities (and thus remote eligibility) and compensation. By adjusting for these structural and individual factors together, we aim to reduce bias from systematic differences between remote and in-office workers.

The Stack Overflow data is well-suited because it contains these variables needed to adjust for such differences. The survey's large sample size also helps in finding comparable groups; for instance, in the 2023 survey, nearly 44% of developers reported being fully remote, and about 14% fully in-person.

One important challenge in our dataset is that the 2022 Stack Overflow survey does not include the 'Industry' field, which appears in the other datasets. Industry is almost certainly an important confounder, as both the treatment and the outcome variables can be greatly influenced by it. However, as eliminating the 2022 survey from our dataset and research is something we wish to avoid, we will use it and address the issue of the missing data and potential solutions in section 4.

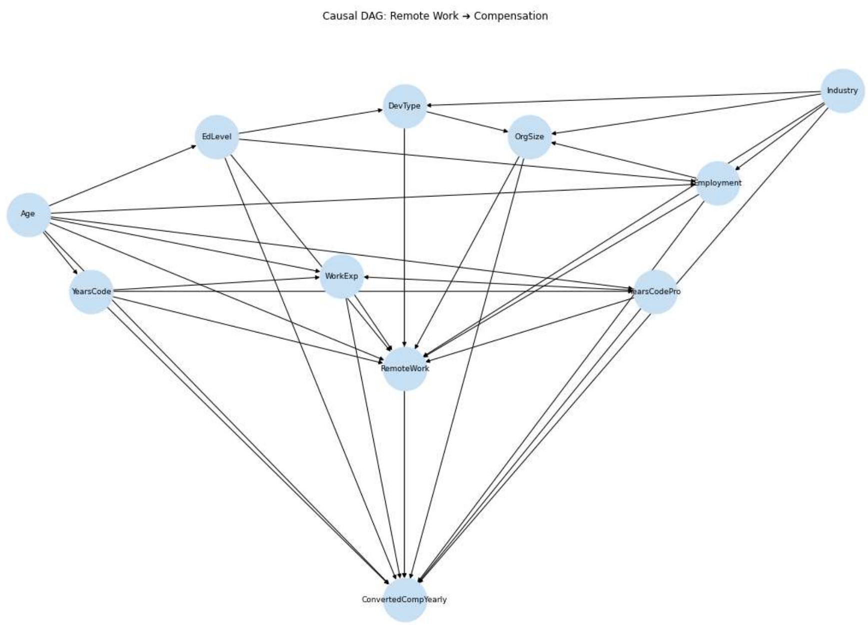
4. Estimation Methods and Assumptions

4.1 Estimation Methods

First before starting our estimation methods, we complete the missing data of Industry column from the year 2022. We trained a Random Forest Classifier using the rows with known industry labels. Categorical features and the target variable were label-encoded, and the data were split into training and validation sets. The model achieved a high validation accuracy of **97.1%**, with strong precision and recall across most industry categories. While a few smaller classes like Energy, Legal Services, and Oil & Gas showed lower performance, the overall results were robust. We then used the trained model to predict and fill in the missing industries, and saved the completed dataset for further analysis.

1. Propensity Score Matching: First, we fit a propensity model to evaluate the probability for being in the treatment group (i.e. work remotely) for each individual in our dataset, using XGBoost and Logistic Regression. We will evaluate the quality of our model using a calibration curve and the Brier score. We used trimming (lower trim of 0.08, and upper trip of 0.97) to enhance balance between treatment and control groups and to avoid biased results.
2. S-learner: To estimate the treatment effect, we tried to use the S-learner with XGBoost, linear regression, and Gradient Boost regression for our predicting functions.
3. T-leaner: Similarly to the S-learner method, we also used the T-learner method as an alternative. In this method, we trained two separate models (i.e. linear regression), one for treatment and one for effect, each predicting the salary based on the data points. We then estimated the treatment affect as the difference in the predictions of the two models. We used the same predicting functions as we did in the S-learner method, which are XGBoost, Gradient Boosting and linear regression.
4. IPW-weighted S-learner : This is a method for estimating treatment effects using a single predictive model that incorporates both the treatment indicator and covariates. It works by first estimating propensity scores-the probability of receiving treatment given covariates using them to compute inverse probability weights (IPW). These weights adjust for differences in covariate distributions between treated and control groups. The outcome model is then trained on the entire sample, but weighted so that underrepresented groups receive more influence. This is especially important in cases of treatment imbalance, where one group (e.g., treated or control) dominates the data, potentially biasing the model. By reweighting the data, the IPW-S-learner reduces this bias, helping to mimic a randomized experiment and produce more reliable estimates of the average treatment effect (ATE).

In all methods, we quantified the uncertainty in our causal effect estimates using confidence intervals.



4.2 Causal Assumptions

Pearls DAG

We have mapped our causal assumptions to this causal graph (Figure 1).

SUTVA

We assume SUTVA holds approximately, meaning developers' salaries are not affected by coworker's remote or in-person status beyond what is already captured by observed covariates such as company size or industry. Our sample consists of individuals from many different companies and roles, which reduces the likelihood of direct interference. The treatment is defined unambiguously as fully remote (100% off-site) versus fully in-person (0% remote), excluding hybrid arrangements to avoid hidden versions of treatment.

Consistency

We assume consistency, namely that the reported salary for a developer observed as fully remote equals that developer's potential salary under remote work, and likewise for fully in-person. This is plausible because the Stack Overflow survey collects the current work arrangement and current annual compensation together, so we can reasonably assume that the salary reported is the salary under the reported work mode. We further enforce it by applying a strict binary treatment definition and excluding implausible or inconsistent

Figure 1

responses.

ignorability

Our identification relies on ignorability given X, where X includes work experience, education level, developer type, organization size, years of coding, industry, employment type, and age. These covariates capture the main determinants of both remote status and salary documented in prior literature. Under this assumption, remote status is independent of potential salaries after adjustment. We assess plausibility by checking covariate balance after matching/weighting and by conducting placebo and sensitivity analyses to gauge robustness to potential unmeasured confounding.

Positivity/Overlap

We assume positivity within the analyzed sample, meaning that for all covariate profiles in our target population, there is a non-zero probability of being in either treatment group. While the original data are imbalanced (85% remote vs. 15% in-person), the large sample size ensures that most covariate combinations appear in both groups. We first assess the reliability of our propensity score model through a calibration plot (Figure 3), which shows that predicted probabilities from the logistic regression model align closely with observed treatment frequencies (The Brier Score for logistic regression is 0.106, and for XGBoost is 0.1105). We then examine the distribution of estimated propensity scores (Figure 2), which reveals substantial, though not perfect, common support between the remote and in-person groups. If necessary, we will restrict to the region of common support and consider trimming or stabilized weights to avoid extreme extrapolation.

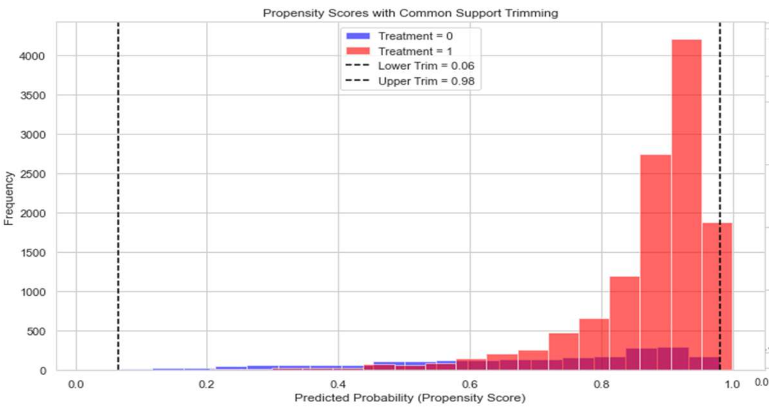


Figure 3

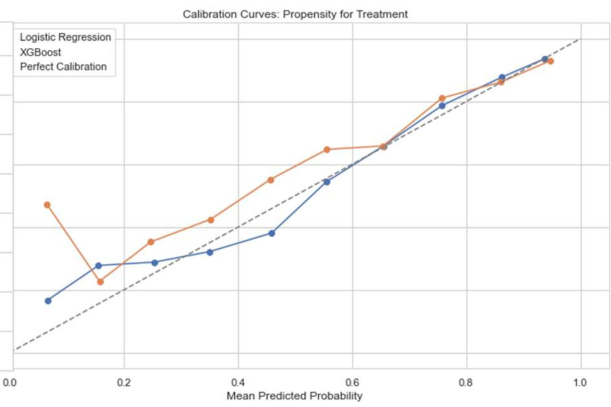


Figure 2

5. Results Analysis

5.1 Propensity Score Matching, Calibration, and Trimming

Brier Scores

- Logistic regression: *0.1059*
- XGBoost: 0.1105
- Trimming bounds: After trimming scores outside the [0.06, 0.98] range, only 1.5% of the sample was dropped, keeping 14173 / 14389 rows in the data, suggesting good common support.

S-learner Results

The S-learner method attempts to estimate treatment effects by fitting a single predictive model for the outcome using both covariates and the treatment indicator as input. Once trained, this model will be used to predict the outcome for each individual twice - once with treatment ($T = 1$) and once for $T = 0$. The difference between these two predictors is the estimated treatment effect.

Model Selection (RMSE):

- Linear Regression: 93,850
- GradientBoostingRegressor: 94,094
- XGBRegressor: 99,312

Observations About Fit

- RMSEs are very large numerically, which is normal for annual salary in dollars. What matters is relative fit across models.
- Linear Regression got a slightly better RMSE than Gradient Boosting (~244), and XGBoost severely underperforms. This could suggest that the nonlinear patterns, if they exist, don't explain much additional variance in salary.

ATE Estimates from the S-learner:

- Point Estimate: \$6,455
- Bootstrap:
  - ATE: \$7,653
  - 95% CI: $[-3{,}969\ ,\ 16{,}156]$

Uncertainty:

- Wide CI implies high residual variance in salaries and/or limited effective overlap.
- The bootstrap ATE being larger than the point estimate from the base S-learner can happen if the bootstrap reweighting reduces influence of regions where the model extrapolates or if the S-learner's base estimate was computed on a slightly different subset.

IPW-weighted S-learner :

This is a method for estimating treatment effects using a single predictive model that incorporates both the treatment indicator and covariates. It works by first estimating propensity scores-the probability of receiving treatment given covariates using them to compute inverse probability weights (IPW). These weights adjust for differences in covariate distributions between treated and control groups. The outcome model is then trained on the entire sample, but weighted so that underrepresented groups receive more influence. This is especially important in cases of treatment imbalance, where one group (e.g., treated or control) dominates the data, potentially biasing the model. By reweighting the data, the IPW-S-learner reduces this bias, helping to mimic a randomized experiment and produce more reliable estimates of the average treatment effect (ATE).

Model Selection (RMSE):

In the IPW-weighted S-learner setup, we compared four models: HistGradientBoostingRegressor, RandomForestRegressor, Ridge, and Lasso, evaluating their predictive performance using cross-validated

RMSE. **Ridge** regression achieved the best performance (RMSE: 677,652), slightly ahead of HistGradientBoosting, and was therefore selected as the final base learner. Unlike our regular S-learner implementation, we did not include XGBoost or GradientBoostingRegressor here because these models have incomplete or inconsistent support for sample weighting, which is crucial in the IPW framework. In contrast, the models we used integrate more reliably with scikit-learn's sample_weight parameter, ensuring proper reweighting during training.

ATE and CI results:

- ATE (point estimate): 29100.5101
- 95% CI: [5483.4841, 51559.7891]

Our results suggest that remote work is associated with an increase in yearly salary of approximately $29,100, on average. The 95% confidence interval ranges from $5,483 to $51,560, indicating that the effect is statistically significant and economically meaningful, as the entire interval lies above zero. This implies that, after adjusting for observed confounders using inverse probability weighting, remote work appears to have a substantial positive impact on annual earnings.

T-learner Results

Unlike the S-learner, which fits a single predictive model, the T-learner fits two predictive models - one for treatment and one for control. To estimate the treatment effect, the T-learner applies both models to each individual and computes the difference. This method is more flexible in capturing heterogeneous effects because it allows the treatment's and control's predictive functions to differ, but it can also be less stable when overlap between groups is limited.

Model Selection Results for T-Learner

For the T-learner, separate models were trained for the treatment group (T=1) and the control group (T=0). Model performance was evaluated using RMSE on validation data:

Treatment group (T=1):

- Linear Regression: RMSE = 600,389
- Gradient Boosting: RMSE = 575,907
- Random Forest: RMSE = 670,719

Control group (T=0):

- Linear Regression: RMSE = 366,449
- Gradient Boosting: RMSE = 354,350
- Random Forest: RMSE = 339,044

These results indicate that Gradient Boosting was best suited for predicting outcomes in the treated group, while Random Forest performed best for the control group. No single model dominated across both groups, which justifies the T-learner strategy of choosing the best-performing model separately within each group.

Estimated Treatment Effect (ATE)

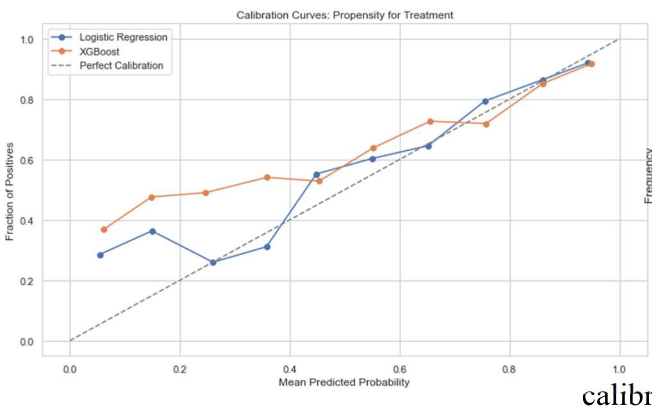
- ATE: $63,354.81
- 95% Confidence Interval: [$58,047.25, $69,167.63]

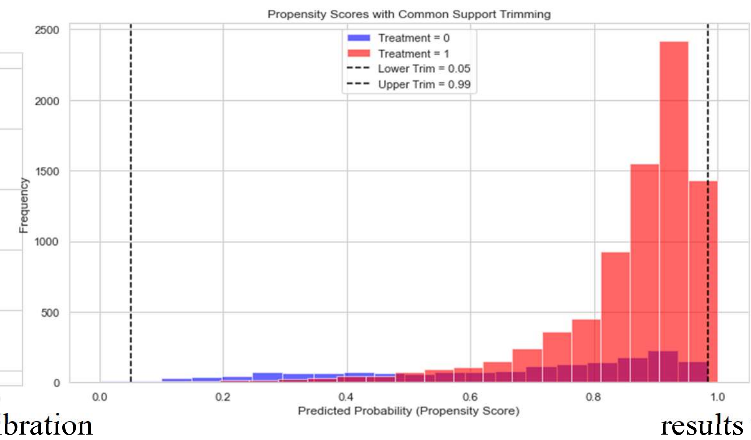
The estimated effect of remote work on annual salary is substantial and statistically significant, as the entire confidence interval lies well above zero. This suggests that, after accounting for observable differences between the groups, remote work is associated with an average salary increase of over $63,000 per year.

6. Robustness Checks Analysis

6.1 Subset Analysis

- **2023–2024 restricted sample**: in this subset, we use only years 23-24, excluding the data from 2022.





The calibration results indicate that logistic regression achieved better calibration (Brier score = 0.1194) than XGBoost (0.1288), making it the preferred propensity score model. The overlap range is reasonably wide (0.05–0.99), and only 1.6% of observations were trimmed, leaving 9,545 rows out of 9,702 for the causal analysis. This suggests sufficient common support while retaining nearly all of the sample.

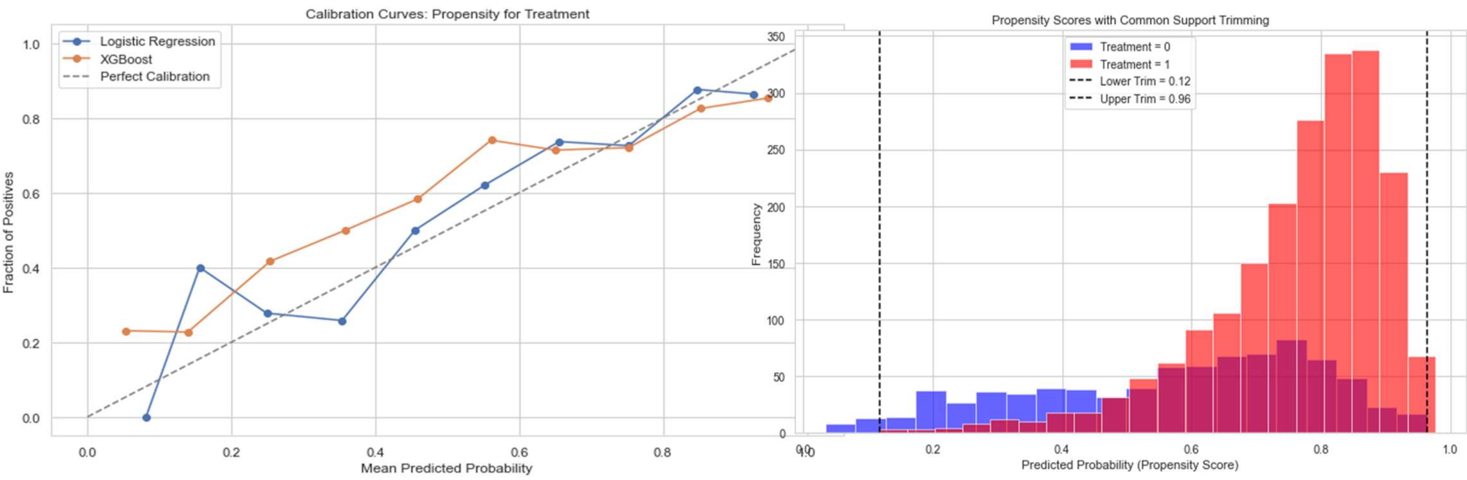
- T- learner $ATE = 8874.1951(95\%\ CI\ [6688.1994, 11005.7390])$
  In the restricted 2023–2024 sample, model selection for the T-learner identified GradientBoostingRegressor as the best fit for the treatment group (RMSE ≈ 94,996) and RandomForestRegressor for the control group (RMSE ≈ 112,663). Using these models, the estimated average treatment effect (ATE) of remote work on annual salary is about $8,874, with a 95% confidence interval of [$6,688, $11,006]. The confidence interval lies entirely above zero,

indicating a **statistically significant and positive effect** of remote work on salaries in this recent sample.

- *IPW-S-learner ATE = 4360.8899 (*(95% *CI [-362.0191, 12575.8486])*

  For the IPW-weighted S-learner in the restricted 2023–2024 sample, model selection based on cross-validated RMSE favored the HistGradientBoostingRegressor (CV RMSE ≈ 95,646), which slightly outperformed Random Forest, Ridge, and Lasso. Using this model, the estimated average treatment effect (ATE) of remote work on salary is about $4,361, with a 95% confidence interval of [−$362, $12,576]. While the point estimate suggests a positive effect, the confidence interval includes zero, indicating that the result is less precise and **not statistically significant** at conventional levels.

- **<u>Junior developers</u>**: in this subset we defined seniors as those who have at least 5 years of pro coding experience and juniors are those who have less.

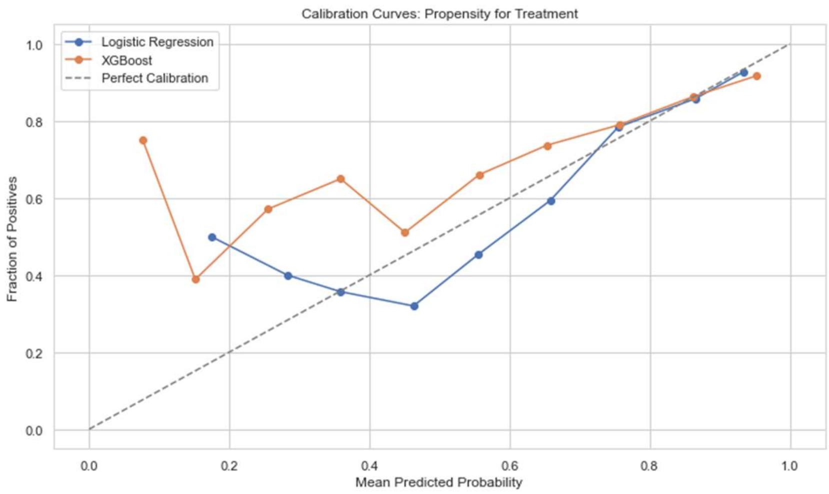


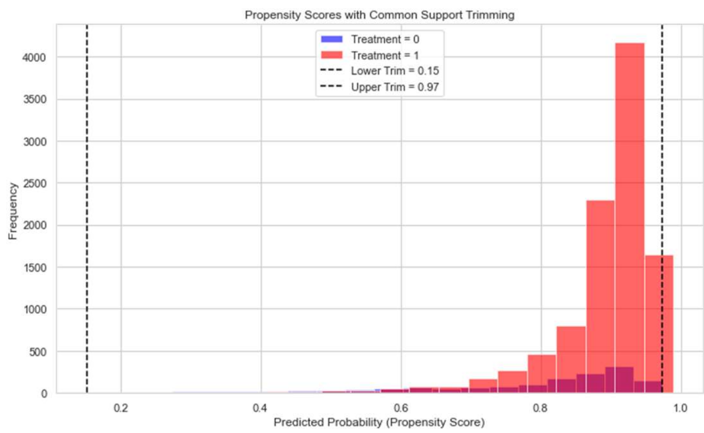
For the Juniors subset the calibration results show that logistic regression (Brier score = 0.1710) outperformed XGBoost (0.1782), making it the preferred propensity score model. The overlap range is fairly broad (0.12-0.96), indicating good common support between treated and control groups. Only 0.9% of observations were trimmed, leaving 2,796 out of 2,822 rows for the causal analysis, which ensures that nearly the entire sample is retained while maintaining overlap quality.

- T-learner $ATE = 67398.9509$ (95% CI [53234.4585, 82330.2404])
  For the junior developers' subset, model selection for the T-learner identified LinearRegression as the best model for the treatment group (RMSE ≈ 653,971) and GradientBoostingRegressor as the best model for the control group (RMSE ≈ 349,975). Based on these models, the estimated average treatment effect (ATE) of remote work on yearly salary is approximately $67,399, with a 95% confidence interval of [$53,234, $82,330]. The confidence interval lies well above zero, indicating a statistically **significant and substantial positive effect** of remote work for junior developers.

- *IPW-S-learner ATE = 21806.9366* ((95% CI *[-26941.3778, 59332.7524]*)
  For the junior developers' subset, cross-validation results showed that the HistGradientBoostingRegressor achieved the lowest RMSE (≈ 564,353), slightly outperforming Ridge and clearly better than Random Forest and Lasso, making it the selected model for the IPW-weighted S-learner. Using this model, the estimated average treatment effect (ATE) of remote work on annual salary is about $21,807, but the 95% confidence interval [−$26,941, $59,333] is wide and includes zero. This indicates that while the point estimate suggests a positive effect, the result is **not statistically significant**, reflecting high uncertainty and potential instability in the junior subgroup.

- **Senior developers**: in this subset we defined seniors as those who have at least 5 years of pro coding experience.



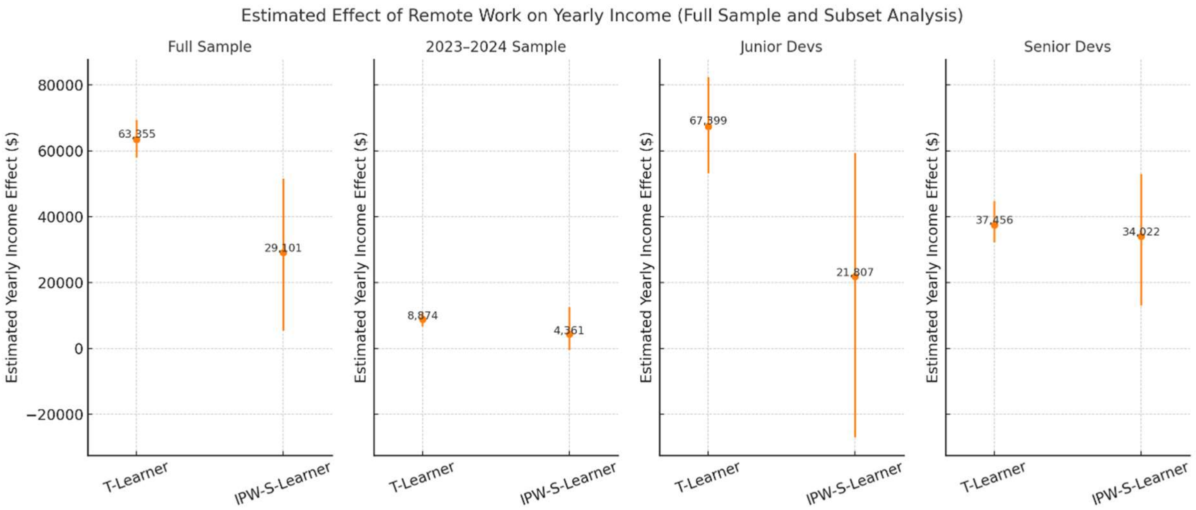


The calibration results show that logistic regression achieved a lower Brier score (0.1000) compared to XGBoost (0.1068), making it the better-calibrated propensity score model. The estimated overlap is fairly strong, ranging from 0.15 to 0.97, which indicates good common support between treated and control groups. Only about 1.2% of the data were trimmed, leaving 11,424 out of 11,567 rows for the causal analysis, ensuring that almost the entire sample is retained while maintaining overlap quality.

- T-learner ATE: 37455.8793(95% CI [32122.1121, 44724.9338])
  For the senior developers' subset, model selection for the T-learner identified GradientBoostingRegressor as the best model for the treatment group (RMSE ≈ 1,021,802) and LinearRegression as the best model for the control group (RMSE ≈ 491,323). Using these models, the estimated average treatment effect (ATE) of remote work on yearly salary is approximately $37,456, with a 95% confidence interval of [$32,122, $44,725]. Since the confidence interval is well above zero, this provides strong evidence of a **statistically significant and substantial positive effect** of remote work for senior developers

- *IPW-S-learner ATE = 34021.6019 (95% CI [13007.3914, 52927.1170])*
  For the senior developers' subset, cross-validation identified Ridge regression as the best-performing model (CV RMSE ≈ 604,194), slightly outperforming Lasso and HistGradientBoosting, and clearly better than Random Forest. Using this model in the IPW-weighted S-learner, the estimated average treatment effect (ATE) of remote work on annual salary is about $34,022, with a 95% confidence interval of [$13,007, $52,927]. The confidence interval lies entirely **above zero, indicating a statistically significant and robust positive effect** of remote work for senior developers.

In conclusion:



In conclusion, the analysis shows that remote work is generally associated with higher income, though the strength and precision of the effect vary across groups. The positive impact appears weaker in 2023–2024 and is highly uncertain among junior developers, where results differ substantially between methods. By contrast, the evidence for senior developers is consistent and statistically significant across approaches, indicating a robust income premium from remote work for more experienced workers. Overall, the findings suggest that while remote work can raise earnings, the effect is most reliable and credible for senior developers.

Placebo analysis:

- The placebo test using **education level** (edlevel) showed an ATE of 0.0004 with a 95% CI of [0.0001, 0.0007], indicating no meaningful effect, just as expected.
- Using **age** as the outcome, the ATE was -0.00004 with a 95% CI of [-0.0004, 0.0006], again showing no evidence of a treatment effect.
- For **YearsCode**, the ATE was 0.00005 with a 95% CI of [-0.00007, 0.00015], also consistent with no effect.
- Using **YearsCodePro** as the outcome resulted in an ATE of -0.00012, with a 95% confidence interval of [-0.00023, 0.00002]. also consistent with no effect.
- using **OrgSize** as the outcome produced an ATE of approximately -0.00006, with a 95% confidence interval of [-0.00035, 0.00066]. The estimate is close to zero and the interval includes zero, indicating no significant effect.
-

All five results support that the model is not capturing spurious effects, reinforcing the validity of the main causal findings.

7. Discussion

Conclusions and Confidence in the Results

Across multiple causal estimation methods, our analysis consistently finds a positive effect of full-time remote work on software developers' salaries. In the full 2022–2024 sample of U.S. developers, remote workers earn substantially more on average than comparable in-office workers after adjusting for observed covariates. For example, using an IPW-weighted S-learner, we estimated an average treatment effect (ATE) of about $29,100 higher annual compensation for remote developers, with a 95% confidence interval of roughly $5,500 to $51,500. This entire interval lies above zero, indicating a statistically significant and economically meaningful premium for remote work. A standard T-learner approach, which allows more flexible outcome models for each group, produced an even larger estimated ATE of about $63,000 (CI ≈ $58k–$69k), likewise suggesting a significant salary increase from remote work. The magnitude of the effect varies by method – the T-learner yielded roughly double the estimate of the IPW S-learner – which signals some sensitivity to modeling assumptions. However, all methods agree on the direction of the effect (a positive premium), bolstering our confidence that fully remote developers tend to earn *more* than observationally similar in-office peers.

We also conducted several robustness checks to assess the stability of these findings. First, restricting the analysis to the more recent 2023–2024 data (excluding 2022) yields a smaller estimated effect: around $8,874 higher salary for remote work according to the T-learner (95% CI $6,688–$11,006) and about $4,361 using IPW S-learner (CI spanning –$362 to $12,576, thus not statistically significant). In other words, with only the post-pandemic years, the point estimates still indicate a positive remote premium, but the effect size is modest (~$4k–$9k) and less precisely estimated in some specifications. This attenuation suggests that the remote wage premium might have narrowed in the most recent period, or it could reflect reduced sample size and variability. Second, stratifying by experience reveals interesting heterogeneity. Among senior developers (≥5 years experience), the remote wage boost is consistently strong and significant across methods. For instance, seniors see an average gain of $37,000 (CI ~$32k–$45k) under a T-learner, and $34,000 (CI ~$13k–$53k) with IPW S-learner. This alignment across methods increases confidence that more experienced developers realize a substantial and robust pay increase from remote work. In contrast, for junior developers (<5 years experience), the results are mixed. The T-learner indicates a very large effect (≈$67,400 higher, CI $53k–$82k), but the IPW S-learner finds a smaller and statistically insignificant effect ($21,800, CI spanning –$26k to $59k). Such divergence suggests the junior subgroup estimates are less reliable, potentially due to model instability or limited data. Overall, our findings imply that *on average* remote work is associated with higher pay, but the credibility of the effect is strongest for senior developers (where results were consistent and precise) and more tentative for junior developers and the most recent cohort.

Importantly, we undertook placebo tests to check for spurious effects. We tested "fake outcomes" that remote work should not causally influence – such as education level, age, and years of coding experience – expecting no effect. Indeed, all placebo ATEs were essentially zero (e.g. an estimated effect on education was 0.0004 with 95% CI [0.0001, 0.0007]). None of these confidence intervals indicated any meaningful difference, which reassures us that our models are not simply picking up spurious correlations or artifacts. This strengthens confidence that the positive salary effects we estimate are not due to an ill-specified model accidentally "predicting" changes in immutable traits. Combined with checks on propensity score calibration and overlap, these diagnostics suggest the analysis is internally consistent. However, caution is still warranted in interpreting the magnitude of the causal effect. The fact that different estimation strategies (and subsets) yield different ATE sizes – from a few thousand dollars up to tens of thousands – indicates that the exact premium is uncertain and sensitive to methodology. We can be fairly confident in the qualitative conclusion that fully remote work tends to correlate with higher compensation for software developers (especially experienced ones), but less certain in the precise magnitude of that causal premium.

Limitations

While we attempted to rigorously control for confounders, this study has important limitations. The most significant one is the assumption of ignorability. Our identification strategy assumes that we observed all the key factors that influence both the likelihood of working remotely and the salary. We included many such covariates – years of experience, education, role, industry, company size, employment type, age, etc. – but it is plausible that *unobserved variables* remain. For example, it may be that only the most skilled or indispensable developers are allowed to go fully remote (e.g. a senior engineer whom the company doesn't want to lose). Such "critical" talent or negotiation ability is hard to quantify and likely was not fully captured by our survey variables. If so, remote status would be correlated with innate ability or other omitted traits, violating ignorability and inflating the estimated effect. In other words, high-performing developers might both negotiate remote arrangements and command higher pay, which could make remote work look beneficial even if it confers no causal raise. We tried to mitigate this by controlling for proxies like experience and job type, but residual bias could remain.

Another limitation is the imbalance between groups: about 85% of the sample was remote versus 15% in-office, leaving relatively few comparators. Trimming extreme propensity scores improved overlap, but in some subgroups (e.g., juniors or niche industries) models may still extrapolate, which likely explains why the T-learner produced inflated effects for juniors while the IPW S-learner was more conservative. Outcome variability also poses risks, as salaries vary widely and a few very high earners could disproportionately influence results. While bootstrap intervals and cross-validation help, testing multiple learners introduces some model selection bias. In addition, the Stack Overflow survey is self-reported and not random, potentially skewed toward higher-earning or remote-friendly respondents, and the 2022 survey lacked industry data, which we imputed with high but imperfect accuracy. Finally, the scope is limited to U.S. software developers during pandemic-era conditions, excluding hybrid workers and firm-level effects. Taken together, these issues mean our estimates should be interpreted as suggestive rather than definitive evidence of a causal premium for remote work.

Comparison with Prior Literature

Our findings both align with and diverge from the emerging literature on remote work and compensation. On one hand, the qualitative pattern – remote workers earning higher wages on average than otherwise similar in-office workers – is consistent with prior studies. A U.S. Bureau of Labor Statistics study using 2020–2021 data found that remote workers enjoyed about a 13% wage premium relative to on-site workers after controlling for many observables. That study applied OLS with rich controls and concluded the premium was robust, suggesting remote work itself might boost wages. Our main full-sample estimate of ~20% (or tens of thousands of dollars) is larger in magnitude, but our 2023–2024 subset result (~$8k–$9k premium) corresponds to roughly 10–15% of an average developer salary, which is quite comparable to the ~10% premium reported in those earlier works. This suggests that by 2023, the remote wage differential in our data converged toward the same ballpark as other findings. Differences in magnitude could be partly due to sample and context. the BLS study looked at the broad workforce (across many occupations), whereas our sample is exclusively software developers, who generally have higher salaries and may have experienced a different labor market dynamic during the pandemic. It is plausible that at the height of the tech talent crunch (2021–2022), remote tech jobs commanded an especially large premium – or that remote roles were more common at top-paying tech firms – inflating the observed effect in our full sample. By 2023, as tech hiring cooled, the premium may have shrunk, which is reflected in our yearly subset analysis.

On the other hand, not all studies frame remote work as a straightforward wage booster. Some highlight that employees value flexibility so much they may accept slightly lower pay in exchange. Barrero et al. (2022), for instance, show that many employers consciously expanded remote work options to ease wage-growth pressures—essentially treating flexibility as part of the compensation package. They estimate that about 38% of firms used remote work this way, which translated into wage growth being about two percentage points lower over two years than it might otherwise have been. In other words, from the employer's perspective, remote work wasn't just a perk layered on top of pay, but a substitute for part of it.

Bibliography

[1] J. M. Barrero, N. Bloom, S. J. Davis, B. Meyer and E. Mihaylov, "The Shift to Remote Work Lessens Wage-Growth Pressures," Becker Friedman Institute for Economics, Chicago, 2022.

[2] U. D. o. Labor, U. B. o. L. Statistics and O. o. P. a. Technology, "Remote Work, Wages, and Hours Worked in the United States," 2024.