Final Report

Cuthbert Chow, Rong Li, Andy Yang

2021-12-03

Contents

Aim and Summary

One of the most important things in the job search is about the salaries, specifically, does this job's salary meet our expectations? However, it is not that easy to set proper expectations. Setting an expectation too high or too low will both be harmful to our job search.

So, the main predictive question we wish to answer is what we can expect a person's salary to be in the US, given a certain professional history (such as years of experience, industry, or age). We will use a linear regression model to do the prediction. In the process, we wish to understand which factors provide the most predictive power when trying to predict a person's salary.

Methods

The Python (Van Rossum and Drake 2009) and R (R Core Team 2021) programming languages and the following Python and R packages were used to perform the data analysis and present results: Pandas (Reback et al. 2020), Scikit-learn (Pedregosa et al. 2011), Altair (VanderPlas et al. 2018), docopt (Keleshev 2014), knitr (Xie 2021).

As references, we utilized (Jain 2017) for methodological practices regarding linear, ridge and lasso regression, as well as (Martín et al. 2018) which recommended linear regression for problems similar to the one we are analysing.

Results & Discussion

Data

The dataset we are analysing comes from a salary survey from the "Ask a Manager" blog by Alison Green. This dataset contains survey data gathered from "Ask a Manager" readers working in a variety of industries. (Green 2021)

We used Altair (VanderPlas et al. 2018) to create figures, Pandas (Reback et al. 2020) to do data processing, and Scikit-learn (Pedregosa et al. 2011) to perform statistical analysis.

Results and Discussion

First, we looked at the distribution of our target "Annual Salary". As shown in the graph below, it seems to be a largely right-skewed distribution. And the median salary is around $80,000.

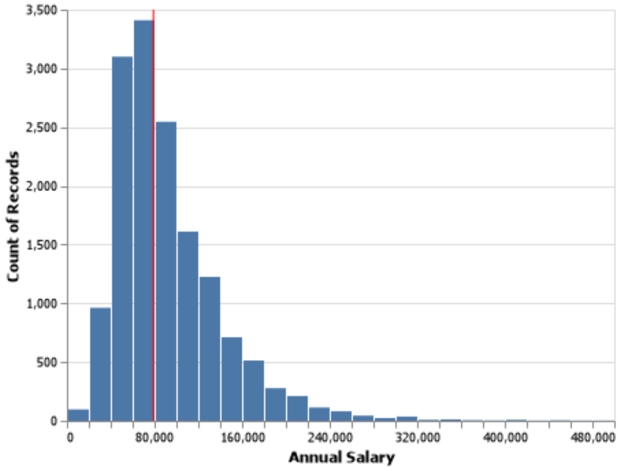

Figure 1: Figure 1 - Distribution of Annual Salaries

Here is some general information about our dataset:

To look at whether the features in our dataset are useful to predict annual salary, we first looked at a summary table about our features:

Table 1: Table 1 - Summary Information About Key Features

Features	Not.Null.Count	Null.Count	Number.of.Unique.Values	Some.Values	Types
how_old_are_you	15037	0	7	['45-54', '25-34', '35-44', '55-64', '65 or over']	object
industry	15008	29	675	['Accounting, Banking & Finance', 'Engineering or Manufacturing', 'Education (Higher Education)', 'Computing or Tech', 'Health care']	object
job_title	15037	0	7970	['CPA', 'Sales Analyst 1', 'Director of Enrollment', 'Process Analyst', 'Senior Data Scientist']	object
other_monetary_comp	11282	3755	583	[10000.0, 2700.0, 0.0, 5000.0, 145000.0]	float64
state	14914	123	108	['California', 'Pennsylvania', 'Colorado', 'Virginia', 'Oregon']	object
city	15006	31	2482	['Palm Springs', 'Pittsburgh', 'Fort Collins', 'Arlington', 'Boulder']	object
overall_years_of_professional_experience	15037	0	8	['21 - 30 years', '11 - 20 years', '8 - 10 years', '2 - 4 years', '5-7 years']	object
years_of_experience_in_field	15037	0	8	['8 - 10 years', '5-7 years', '11 - 20 years', '2 - 4 years', '1 year or less']	object
highest_level_of_education_completed	14935	102	6	["Master's degree", 'College degree', 'Some college', 'PhD', 'High School']	object

We noticed that there are lots of null values in the additional information features (additional_context_on_job_title, additional_context_on_income, etc), and some of the variables have a lot of unique values. Therefore, later we dropped the two additional information features and used the bag-of-words model to extract features from text columns like industry and job title.

Here we want to explore those variables that have < 10 unique values and check their distributions and relationships with the annual salary, since variables with 100s or 1000s of distinct values would be harder to visualize in a meaningful way.

As shown below, the higher salaries are roughly associated with the older age groups, the longer experience and the higher education, which indicates those are likely to be good predictors of our target.

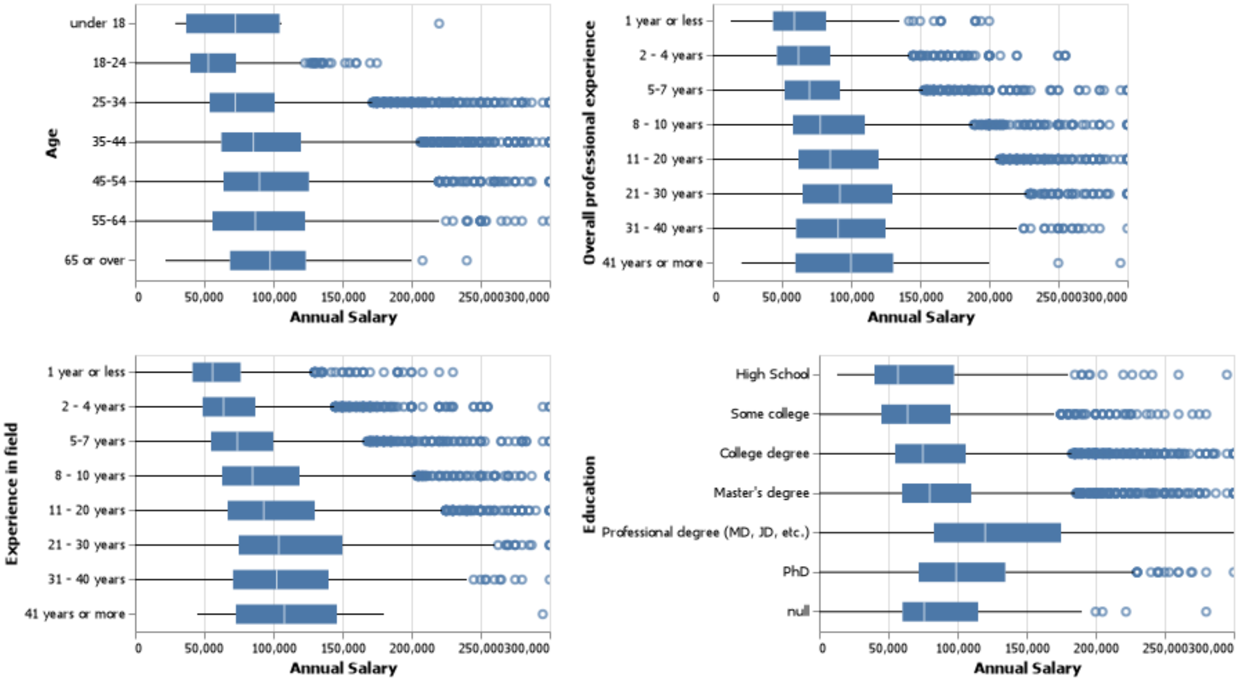

Figure 2: Figure 2 - Salary For Various Categorial Features

We chose two different types of models to predict annual salary based on the given features in the dataset. A linear model, Ridge, and an ensemble model, RandomForestRegressor. To ensure that the models were not overfitting to training data, we conducted some additional data cleaning. Firstly, *annual_salary* values within the training dataset of less than 10,000 USD or over 1,000,000 USD were removed. Additionally, text values that occurred less than 5 times in the *state* or *city* features were imputed with an empty string. This ensures that highly specific values will be removed which ultimately helps reduce overfitting.

To score the models, we relied on the r2 and root mean squared error scores since they are simple to interpret. Since the annual salary target of the test set can be 0, MAPE would not be a suitable metric in this scenario. We did not filter the test dataset to allow for MAPE scoring since this would bias the test set against evaluation data.

Hyperparameter optimization was performed on the Ridge and Random Forest models. For Ridge, the alpha parameter was optimized with a search space spanning $10^{(-5)} - 10^{(5)}$ with 20 total iterations. The ideal alpha value which provided the highest r2 score was determined to be approximately 6.16 as seen by the results table.

Table 2: Table 2.1 - R2 Scores For Various Alpha Values

r2	Negative.RMSE	alpha
0.4952119	-37852.22	6.158482e+00
0.4910222	-38008.79	2.069138e+01
0.4892869	-38074.17	1.832981e+00
0.4768824	-38534.29	5.455595e-01

r2	Negative.RMSE	alpha
0.4740377	-38637.83	6.951928e+01
0.4644786	-38988.93	1.623777e-01
0.4574375	-39244.52	4.832930e-02
0.4530491	-39402.67	1.438450e-02
0.4521830	-39433.41	4.281300e-03
0.4520209	-39439.30	1.129000e-04
0.4517050	-39450.73	1.274300e-03
0.4514334	-39460.59	1.000000e-05
0.4513467	-39463.21	3.793000e-04
0.4509927	-39475.57	3.360000e-05
0.4439409	-39728.05	2.335721e+02
0.4026841	-41175.61	7.847600e+02
0.3457046	-43095.39	2.636651e+03
0.2605887	-45814.10	8.858668e+03
0.1527303	-49041.58	2.976351e+04
0.0661657	-51484.57	1.000000e+05

For Random Forest Regressor, we optimized the n_estimators for speed. We searched for performance increases within the hyperparameters of 10, 20, 50, and 100 trees. We ultimately selected the 50 tree regressor for time savings, since the 100 tree regressor provided very little performance boost compared to processing time required.

Table 3: Table 2.2 - R2 Scores For Various Alpha Values

test.r2	train.r2	Negative.RMSE	n_estimators
0.4586719	0.9250913	-39205.68	100
0.4543887	0.9217410	-39358.31	50
0.4377983	0.8979579	-39947.49	10
0.4341157	0.9156897	-40083.68	20

Using these hyperparameter values, the models were evaluated on the test data. The results can be seen in the table below.

Table 4: Table 3 - Scores of Ridge Model on Test Data

Metric	Ridge.Scores
R2	0.38
RMSE	48398.05

/COMMENT ABOUT THE RESULTS

To visualize the effectiveness of our models, we can plot the predicted salary values against the actual salary values and compare the correlation to a 45 degree line.

COMMENT ON THE GRAPH AND HOW THE MODELS PERFORMED

We can gain insight into how our model makes predictions by analysing the coefficient values associated with the regression. The tables below show the difference in salary that the model predicts given the change in the associated feature for the Ridge model. The first Table displays the top 10 positive coefficients.

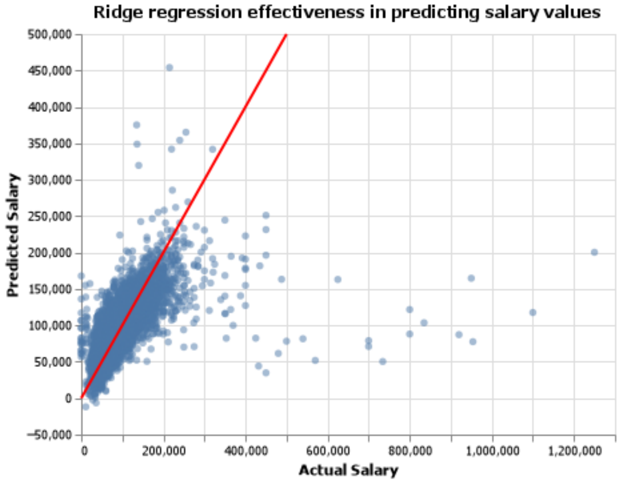

Figure 3: Figure 3 - Actual vs Predicted Salary Values

Table 5: Table 4.1 - Ten most positive coefficients

Feature	Coefficient
physician	74365.20
svp	63705.52
md	62124.66
partner	58462.57
psychiatrist	53442.29
city_Bay Area	46930.74
equity	45417.20
chief	43911.43
machine	41834.97
onlyfans	41535.88

The top 10 most positively correlated features with higher income are somewhat expected, as they mostly consist of text features which represent high-paying jobs, or titles such as MD. An interesting feature we didn't expect was onlyfans, which is a more recent phenomena. This shows the effects of modern technology on methods to earn income.

Table 6: Table 4.2 - Ten most negative coefficients

Feature	Coefficient
paralegal	-38455.38
resident	-28025.23
adjunct	-24879.49
office	-23444.43
clerk	-21626.92
bookkeeper	-20094.96
assistant	-18433.97
city_Tallahassee	-18425.08
legal	-18365.62
secretary	-18257.95

The most negative coefficient features are also somewhat expected, as they mostly consist of traditionally lower paying jobs in the US.

INSERT DESCRIPTION ABOUT COMPARING RIDGE TO RANDOMFOREST (include some text about random forest coefficient values being incomparable between the two)

Table 7: Table 5 - Feature importance comparison

Significance.Rank	Ridge.Feature	Ridge.Coefficient	Random.Forest.Feature	RandomForest.Coefficient
1	physician	74365.20	other_monetary_comp	0.2688
2	svp	63705.52	years_of_experience_in_field	0.0624
3	md	62124.66	highest_level_of_education_completed	0.0519
4	partner	58462.57	computing	0.0501
5	psychiatrist	53442.29	overall_years_of_professional_experience	0.0170
6	city_Bay Area	46930.74	how_old_are_you	0.0135
7	equity	45417.20	state_California	0.0120
8	chief	43911.43	senior	0.0118
9	machine	41834.97	director	0.0115
10	onlyfans	41535.88	education	0.0107

COMMENT ON RESULTS

References

Green, Alison. 2021. "How Much Money Do You Make?" *Ask A Manager*. https://www.askamanager.org/2021/04/how-much-money-do-you-make-4.html.

Jain, Shubham. 2017. "A Comprehensive Beginners Guide for Linear, Ridge and Lasso Regression in Python and r." *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2017/06/a-comprehensive-guide-for-linear-ridge-and-lasso-regression/.

Keleshev, Vladimir. 2014. *Docopt: Command-Line Interface Description Language*. https://github.com/docopt/docopt.

Martín, Ignacio, Andrea Mariello, Roberto Battiti, and José Alberto Hernández. 2018. "Salary Prediction in the IT Job Market with Few High-Dimensional Samples: A Spanish Case Study." *International Journal of Computational Intelligence Systems* 11: 1192–1209. https://doi.org/https://doi.org/10.2991/ijcis.11.1.90.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12 (Oct): 2825–30.

R Core Team. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Reback, Jeff, jbrockmendel, Wes McKinney, Joris Van den Bossche, Tom Augspurger, Phillip Cloud, Simon Hawkins, et al. 2020. *Pandas-Dev/Pandas: Pandas* (version latest). Zenodo. https://doi.org/10.5281/zenodo.3509134.

Van Rossum, Guido, and Fred L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

VanderPlas, Jacob, Brian Granger, Jeffrey Heer, Dominik Moritz, Kanit Wongsuphasawat, Arvind Satyanarayan, Eitan Lees, Ilia Timofeev, Ben Welsh, and Scott Sievert. 2018. "Altair: Interactive Statistical Visualizations for Python." *Journal of Open Source Software* 3 (32): 1057.

Xie, Yihui. 2021. *Knitr: A General-Purpose Package for Dynamic Report Generation in r*. https://yihui.org/knitr/.