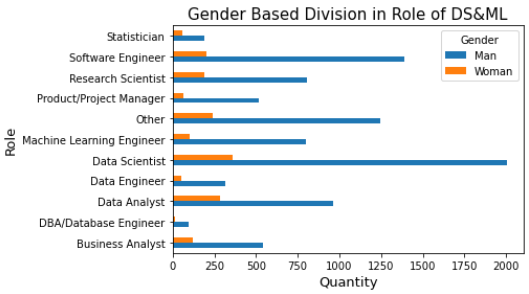
Kaggle ML & DS Survey Challenge

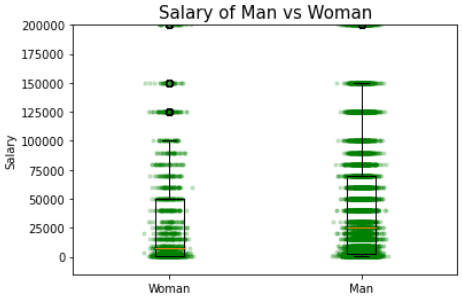
Tell a data story about a subset of the data science community represented in this survey, through a combination of both narrative text and data exploration.

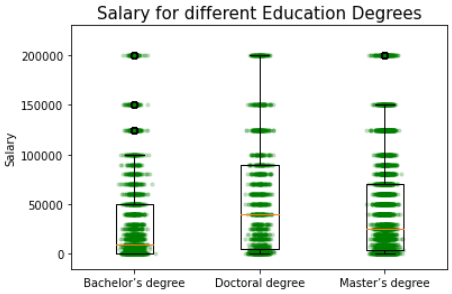
Part 1

Explore the survey data to understand (1) the nature of women's representation in Data Science and Machine Learning and (2) the effects of education on income level.

1.1 Perform EDA to analyze the survey dataset and to summarize its main characteristics





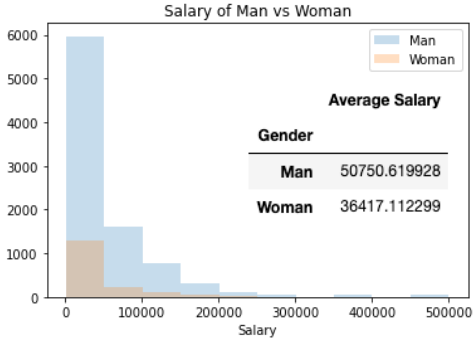


Consider Gender, Education and Salary to explore (1) the nature of women's representation in DS & ML and (2) the effects of education on income level.

The plot '**Gender Based Division in Role of DS&ML**' shows the number of women in all DS&ML roles are far less than man. The plot '**Salary of Man vs Woman**' shows the salary of woman is lower than man in general. The last plot '**Salary for different Education Degrees**' shows the salary for higher degree is greater in general.

1.2 Estimating the difference between average salary of men vs women.

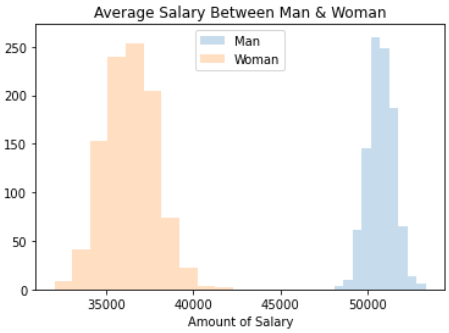
1.2.1 - 1.2.2 Compute descriptive statistics for each group. If suitable, perform a two-sample t-test with a 0.05 threshold.

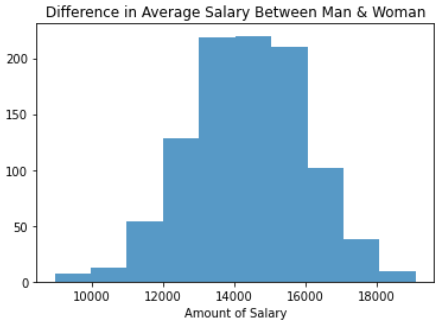


The table in the left contains the average salary of men & women. Applying an independent two-sample t-test is a desired way to test whether the difference of average salary between two gender groups is significant, which assumes (1) Independence (2) Normality (3) Equal variances.

Both distributions have right tails. It's not suitable to apply independent two-sample t-test since the assumption of normality is not met. The sample is skewed, and average salary is largely biased by large values.

1.2.3 Bootstrap your data for comparing the mean of salary for the two groups. Use 1000 replications.





(Bootstrapped Distributions for men and women, and the difference in means)

Adopted bootstrap sample strategy to create larger sample set, also reduce the chance of randomly reject the null hypothesis. (1) First split data by gender, bootstrap data with same size as original data; (2) After that, bootstrap both groups 1000 times, computing mean of salary in each iteration.

1.2.4 If suitable, perform a two-sample t-test with a 0.05 threshold on the bootstrapped data.

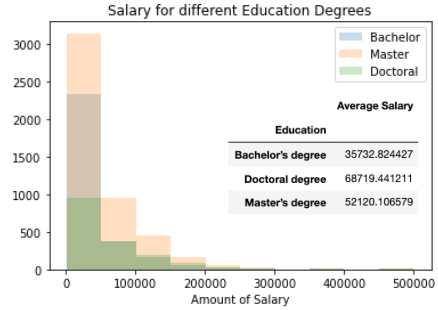
The plots from (c) shows means for man and women are normally distributed.
Perform Levene test for equal variances, $H_0: \sigma^2_{man} = \sigma^2_{woman}$ vs $H_1: \sigma^2_{man} \neq \sigma^2_{woman}$
$p_{value} = 1.60964e-76 < 0.05$, statistically significant => unequal variances
Perform Welch's t-test (not assume equal variance), $H_0: \mu_{man} = \mu_{woman}$ vs $H_1: \mu_{man} \neq \mu_{woman}$
$p_{value} = 0 < 0.05$, statistically significant => unequal means

1.2.5 Findings:

From the two-sample t-test, we could conclude that the average salary difference between two main genders is significant in the field of data science, men tend to have higher salary than women. The difference of average salary is around 14000. But without further analyse, we could not conclude the causation between gender and wage gap. One possibility is there's a mediation factor.

3. Estimating the difference between average salary of different education degree.

1.3.1 – 1.3.2 Compute descriptive statistics for each group. If suitable, perform ANOVA with a 0.05 threshold.

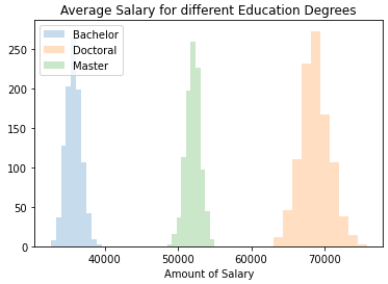


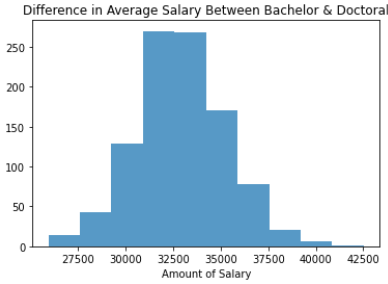
Education	Average Salary
Bachelor's degree	35732.824427
Doctoral degree	68719.441211
Master's degree	52120.106579

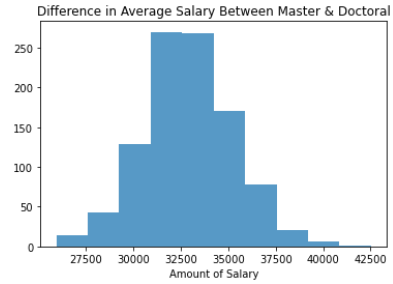
The table in the left contains the average salary of different education degrees. Applying an ANOVA test is a desired way to test whether Bachelor's degree, Doctoral degree, and Master's degree have same average salary, which assumes (1) Independence (2) Normality (3) Equal variances.

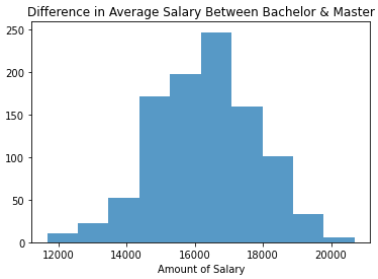
All distributions have right tails. It's not suitable to apply ANOVA test since the assumption of normality is not met. The sample is skewed, and average salary is largely biased by large values.

1.3.3 Bootstrap your data for comparing the mean of salary for the three groups. Use 1000 replications.









(Bootstrapped Distributions for Bachelor's degree, Doctoral degree, and Master's degree, and the difference in means)

Adopted bootstrap sample strategy, the procedure is same as above.

1.3.4 If suitable, perform an analysis of variance (ANOVA) with a 0.05 threshold on the bootstrapped data.

The plots from (c) shows means of all education degree groups are normally distributed.
Perform Levene test for equal variances, $H_0: \sigma^2_{bach} = \sigma^2_{mas} = \sigma^2_{doc}$ vs H_1: at least one variance is different
$p_{value} = 6.27403e-128 < 0.05$, Reject H_0 => At least one variance is different
Perform Kruskal-Wallis ANOVA test (not assume equal variance)
$H_0: \mu_{bach} = \mu_{mas} = \mu_{doc}$ vs H_1: at least one mean is different
$p_{value} = 0 < 0.05$, statistically significant => At least one mean is different

1.3.5 Findings:

From the Kruskal-Wallis ANOVA, we could conclude that the average salary difference among different Education Degrees is significant. Master tends to have higher salary about 16500 than Bachelor, and doctoral tend to have higher salary about 33000 than master. I would suggest perusing a higher degree to have higher salary. However, without further analyze, we could not conclude the causation between education level and wage gap.

Part 2

Train, validate, and tune multi-class ordinary classification models that can classify, given a set of survey responses by a data scientist, what a survey respondent's current yearly compensation bucket is.

2.1 Data Cleaning

Load dataset "clean_kaggle_data_2020.csv" as ***Salaries***, not including the first line describe the questions. Count the number of missing values of features, original dataset has 344 features contain Nan. Prepare clean data for analysis by dealing missing values and convert categorical data into numerical data.

	# Missing Values	% Missing Values
Q34_A_Part_9	10685	99.589897
Q18_OTHER	10681	99.552614
Q28_B_OTHER	10675	99.496691
Q27_B_OTHER	10675	99.496691
Q31_A_Part_9	10673	99.478050
...	...	...
Q15	561	5.228819
Q13	561	5.228819
Q11	561	5.228819
Q8	561	5.228819
Q25	159	1.481965

344 rows × 2 columns

2.1.1 Multiple Choice Question

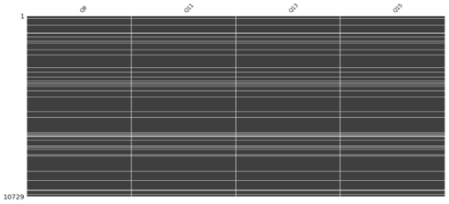
Nan is missing at random and it indicates a survey respondent did not select that given option from a multiple-choice list. First drop the column of choice 'None', since it won't contribute to the salary. Then replace all non-Nan with 1 and fill Nan with 0.

2.1.2 Drop Features

Supplementary and ***Follow-up*** questions are more specific compare to precious questions. Can't treat them equally since they were received by different groups, remove them to avoid add noise to the model. ***Q24*** has been encoded to new futures, drop it to avoid duplicate information. Drop ***Time from Start to Finish (seconds)***, it won't contribute much useful information to estimate salary class.

2.1.3 Drop 561 rows.

The plot of missingno indicates the missing values for ***Q8, Q11, Q13, Q15*** are belong to same respondents.



2.1.4 For features *Q25 and Q38*

Values missing at random, fill with mode. The dataset does not contain any missing value after this step.

	# Missing Values	% Missing Values
Q38	1028	10.110149
Q25	134	1.317860

2.1.5 Ordinal Categorical Features

Encode an interval to the median of the range, encode greater or smaller than a number to the number itself, and remain the same for one number. Noted encoding for ***Q4 (Education)*** is defined differently (encode higher degree to larger number), more details could be found on Jupyter Notebook. This encoding method would reserve the information of order.

2.1.6 Non-Ordinal Categorical Features

Apply dummy encoding. And drop one of each of the dummy variables since it could be implied by the other dummy variable columns. Otherwise, multicollinearity would be a problem. This encoding method would not assume any rank between features.

2.2 Exploratory Data Analysis

Correlation plot (visualize order of feature importance)

Correlation indicates the strength of a relationship between two variables. Features with larger positive correlation with ***Q24_Encoded*** would contribute more to the compensations. And one advantage of using correlation is it's a standardized metric doesn't require feature scaling.

The correlation plot visualizes 20 original attributes most related to a survey respondent's yearly compensation. Some interesting features including:

- Reside in United States
- Code/ programming experience
- Money have spent on ML/cloud at work
- Age
- ML methods experience

(more details on Jupyter Notebook)

Feature engineering is the process to extract useful information from raw data and create features based on the analyzed questions. It's a useful technique to improve Machine learning algorithm performance and predictive accuracy by selecting features with adequate amount of bias and variance.

Standardize Features (mean 0 and std 1)
Features with greater scales would tend to have larger coefficient and would be interpreted as more important features. Standardize features before ordinal logistic regression could avoid this issue to happen.

2.3 Model Implementation and Feature Selection

	0-9,999	10,000-19,999	20,000-29,999
0	Q1 Age (years)	Q1 Age (years)	Q1 Age (years)
1	Q4 Education	Q4 Education	Q4 Education
2	Q6 Code/programming experience (years)	Q6 Code/programming experience (years)	Q6 Code/programming experience (years)
3	Q7_Python	Q7_Python	Q7_Python
4	Q7_R	Q7_R	Q7_R
...	...	...	...
252	Q38 primary tool use at work/school to analyze...	None	Q38 primary tool use at work/school to analyze...
253	Q38 primary tool use at work/school to analyze...	None	Q38 primary tool use at work/school to analyze...
254	Q38 primary tool use at work/school to analyze...	None	None
255	Q38 primary tool use at work/school to analyze...	None	None
256	Q38 primary tool use at work/school to analyze...	None	None

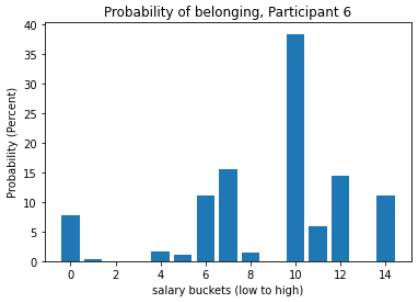
- **Ordinal Logistic Regression**

Start with lowest salary bucket, encode '0-9,999' to 0 and rest buckets to 1. Perform binary logistic regression and get the probability of belonging to bucket '0-9,999'. Then encode '0-9,999' and '10,000-19,999' to 0 and rest buckets to 1. Perform binary logistic regression and get the probability of belonging to class 0. Repeat this process 14 times, we would derive 14 models and probability of belonging to each of 15 salaries buckets.

- **Lasso Regularization (C = 1, penalty = 'l1')**

Using ordinal logistic regression with Lasso regularization and let the algorithm to decide which feature to select. This algorithm basically adds a penalty term l1 norm to the original cost function, the features with coefficient 0 will be dropped and others will be selected. Set **C = 1** for now. C is the inverse of regularization strength. Smaller C values specify stronger regularization, less features would be selected, and vice versa. The table in the left indicates what features have been selected for models that compute probability of belonging to each of the salary buckets. For example, the model compute probability of belonging to salary bucket '20,000-29,999' select 254 features.

The table in the right stores the probability of belonging to each of the salary buckets. The bar plot of probability estimates the salary of participant 6 fall in bucket '100,000-124,999'. But the actual salary bucket is '125,000-149,999'. The model made the wrong prediction for this observation.



	0-9,999	10,000-19,999	20,000-29,999	30,000-39,999
0	3.48	0.80	0.00	1.71
1	72.90	12.08	7.75	2.94
2	2.81	0.00	1.09	0.00
3	8.92	0.00	6.38	9.36
4	1.30	2.40	1.26	5.50
...	...	...	...	...
3046	77.22	17.23	0.00	0.83
3047	21.39	25.57	24.02	0.62
3048	45.00	25.17	14.68	5.44
3049	62.99	22.05	4.45	7.09
3050	90.86	5.33	2.37	0.57

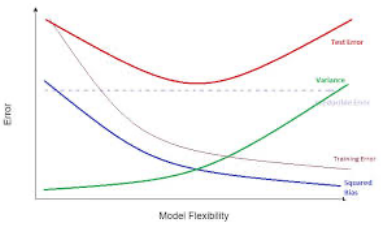
```
Fold 1 accuracy: 0.38764044943820225
Fold 2 accuracy: 0.42275280898876405
Fold 3 accuracy: 0.40870786516853935
Fold 4 accuracy: 0.4157303370786517
Fold 5 accuracy: 0.41151685393258425
Fold 6 accuracy: 0.3960674157303371
Fold 7 accuracy: 0.41853932584269665
Fold 8 accuracy: 0.42616033755274263
Fold 9 accuracy: 0.4289732770745429
Fold 10 accuracy: 0.4092827004219409
Mean r2: 0.4125371371229002
Standard Deviation: 0.012343878943288441
```

- **10-fold Cross Validation**

Model accuracies vary across folds since the training data (9 folds) and the validation data (1 fold) are different in each iteration. When C=1, the model has a cross validation score of **41.25%** with a standard deviation of 1.23%. CV score computes average accuracy over 10 folds, it gives better estimation of the testing error, and it didn't miss any information of data.

2.4 Model Tuning

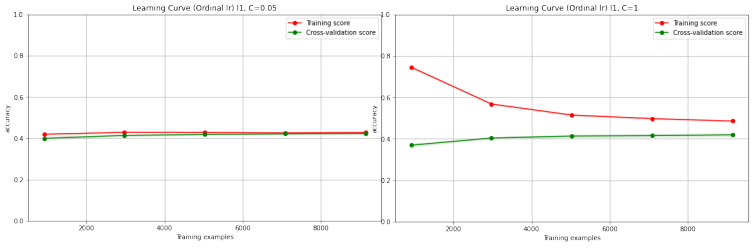
Hyperparameter including penalty (norm), C (inverse of regularization strength), scoring (metric to evaluate performance), cv (number of folds), test size (proportion of test data), etc. Complex model with too many features would have high variance, and also computational expensive. Simple model with too less features would have both high bias and variance. The model has adequate amount of bias and variance, is the one has highest testing accuracy. Set cv = 10 and test size = 0.3. Turn hyperparameter penalty and C and use grid search to find optimal model with the highest CV score. (CV better estimate the testing error)




```
Fold 1 accuracy: 0.4058988764044944
Fold 2 accuracy: 0.448033707865168555
Fold 3 accuracy: 0.42696629213483145
Fold 4 accuracy: 0.4311797752808989
Fold 5 accuracy: 0.41713483146067415
Fold 6 accuracy: 0.398876404494382
Fold 7 accuracy: 0.4339887640449438
Fold 8 accuracy: 0.44866385372714485
Fold 9 accuracy: 0.44866385372714485
Fold 10 accuracy: 0.44022503516174405
Mean r2: 0.4299631394301427
Standard Deviation: 0.016935145872843324
```

Optimal model (C = 0.05, penalty = 'l1')

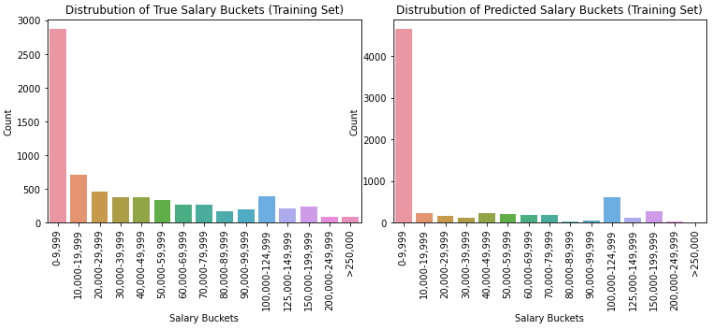
Cross validation score improves to **43.0%** from 41.25%
Testing accuracy improves to **43.13%** from 42.31%



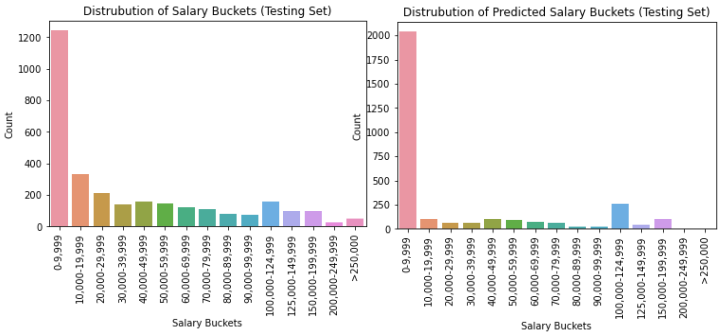
Learning Curve visualize the training and CV score for different training set size. Training score and CV score need to be close enough, otherwise the model would be overfitting and have higher testing error. And both scores have to be large enough to ensure predictive accuracy. C = 0.05 gives better result.

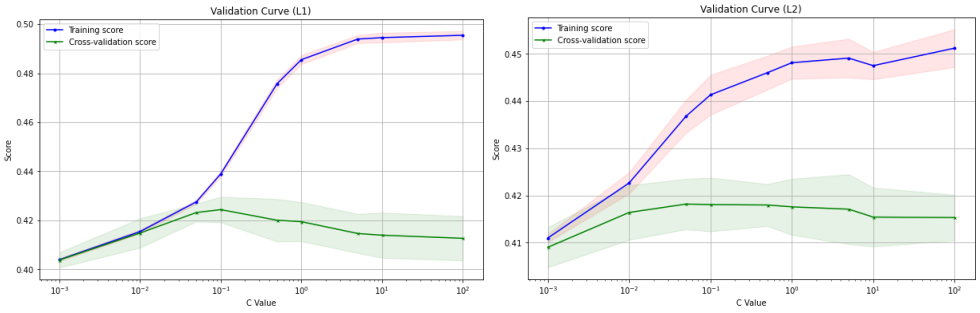
2.5 Testing & Discussion

Apply optimal model with C = 0.05 and penalty = 'l1' to make classifications on the test set. Training accuracy is 47.53% and Testing accuracy is **43.13%**, which imply the model is overfitting a bit. Overfitted model has lower bias but higher variance, higher training accuracy and lower testing accuracy. Model could be improved by decreasing hyperparameter C, the variance and testing error will decrease then. Try C = 0.045, the testing accuracy increased to **43.33%**.



The plots on the right visualize the distribution of true target variable values and their predictions on both the training set and test set. The frequency of predicted salary buckets '0-9,999' and '100,000-124,999' is much larger than the true value. The observations close to those two buckets are more likely to be classified in them. The reason is overfitted model would classify close data together to increase higher variance, but it would decrease accuracy.





Validation Curve visualize how hyperparameter C effect the training and CV score. Add a norm term to model can reduce overfitting. Smaller hyperparameter C specify stronger regularization strength, it can decrease model complexity and variance. Finally, we can achieve better predictive accuracy, although the bias will increase a bit.

From the validation curve, model with C=1 and penalty = 'l1' is optimal.