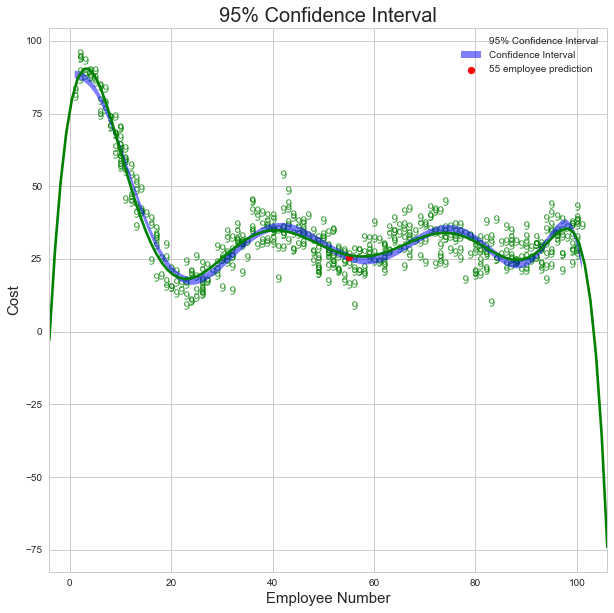
Report: Company Benefit vs Company Number

The company wants to use prediction model to better understand the relationship between company size and corresponding spending on benefits. The manager level is interested in understanding a fifty-five employees’ company. A prediction model is built based on historical data to better predict the average cost on benefits. The cost is between 24.314 and 26.979 at 95% confidence interval. And also we found out for small company (less than twenty employees), average cost will increase dramatically as long as the company fires more employee. However for middle size company (twenty to one hundred employees), the average cost doesn’t change very much.



The up and lower boundary for confidence interval (95%) is shown as blue region around the prediction line. The red point represent the average cost in benefit for company that has 55 employees. The cost is [24.314, 26.979] at 95% confidence interval.

The prediction interval would be wider than confidence interval because we take the true error term into account. By calculation, 95% prediction interval for the average cost for a single company is [ 24.225, 27.068]

Data Exploratory Analysis

Data only has two variables: one is the number of the employees, second variable ‘cost’ is the average cost in the benefits associated with the employees. Since we only have one predictor: employee number. We don’t need to worry about the collinearity between predictors. First step is to understand the data.

*Variables Distribution*

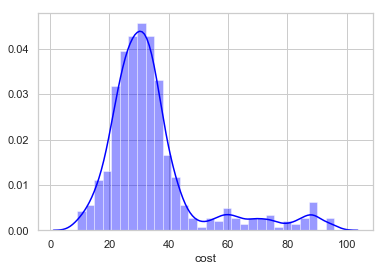
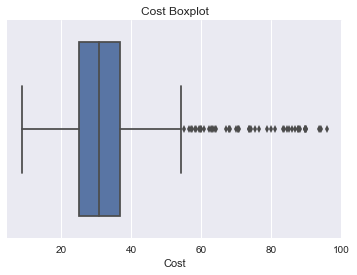
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Count | Mean | STD | Min | 25% | 50% | 75% | Max |
| Employee | 500 | 53.10 | 28.95 | 1 | 30 | 52 | 79 | 101 |
| Cost | 500 | 34.39 | 16.18 | 9.03 | 25.20 | 30.93 | 36.89 | 95.81 |

Both employee and cost are totally 500 each so there is no missing value. Response variable ‘cost’ is a continuing number. Therefore, 1) this is not a binary problem. I will use regression analysis to develop a model. And 2) we will eliminate the logistic regression method due to the continuing number in response variable. Second step is checking the outliers in the existing datasets. Since this is a regression problem, some of the regression methods such as linear regression is very sensitive to the outlier.



Employee Number: Based on predictor boxplot, there is no outliers in the employee number. Majority companies have fifty-three employees.

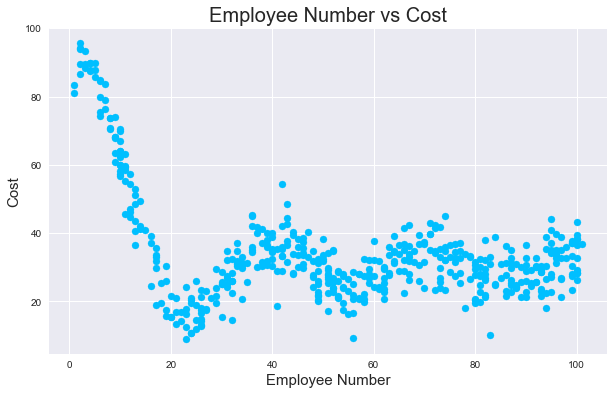
*Cost Outlier Analysis Plot*



Outliers based on ‘Cost’ variable

|  |  |
| --- | --- |
| Employee | Cost |
| 2 | 86.7 |
| 3 | 88.25 |
| 3 | 93.46 |
| …. | …. |
| 7 | 83.716 |

By filtering out the cost value larger than ‘75th Percentile +1.5\*IQR’, I found out nineteen outliers from the cost variable. Also from histogram, the cost distribution is right skewed. However, I don’t want to delete those outliers. Those outliers belong to small number of employees ranging from one to seven per company. At this step, I would use log transformation to deal with the right skewed distribution.



Next, I use scatter plot to visualize the relationship between employee number and cost. The data trend convinces me to try polynomial regression model first. At beginning, the employee cost is expensive as the companies shrinks their sizes. Then the cost tends to be steady (between 20 and 40) even though the company increases its employee pull size.

Modeling

Based on the data exploration, I have applied six major regression models: Lasso, Ridge, Linear Regression, Random Forest and Gradient Boosting. After using cross validation, I rank models based on R\_2 and mean square error.

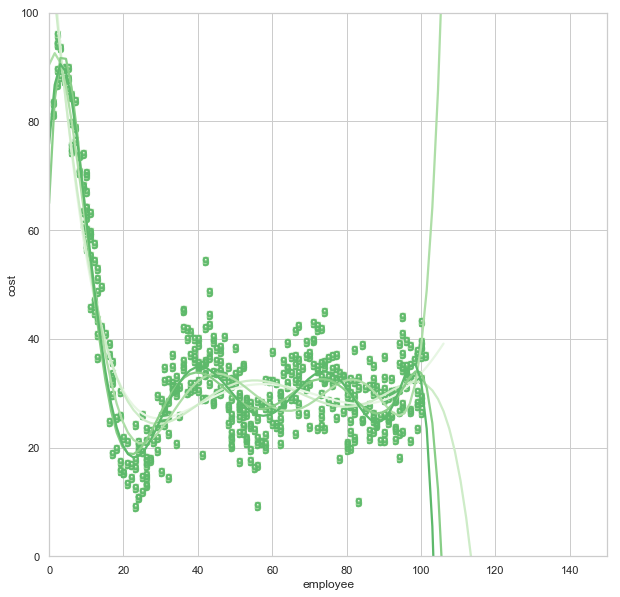
*Model Performance Ranking*

|  |  |  |
| --- | --- | --- |
| Model | R\_2 | Mean Square Error |
| Random Forest | -29.1026 | 176.0 |
| Gradient Boosting | -30.123 | 180.1 |
| Lasso | 0.154708 | 14.073788 |
| Ridge | 0.154458 | 14.075871 |
| Linear Regression | 0.154458 | 14.075872 |
| Polynomial (Degree=3) | 0.355341 | 10.413948 |
| Polynomial (Degree=4) | 0.688000 | 8.31419 |
| Polynomial (Degree=5) | 0.70131 | 8.36108 |
| Polynomial (Degree=6) | 0.702746 | 8.2313 |
| Polynomial (Degree=7) | 0.72355 | 7.38101 |
| Polynomial (Degree=8) | 0.76890 | 7.13449 |
| Polynomial (Degree=9) | 0.83154 | 5.84742 |

It turns out that polynomial regression with nine degrees has the best prediction power: R^2 is 83.154% and mean square error is only 5.84.

I also want to have a visual evaluation on my model result. Therefore, I plotted the polynomial regression models with four different degrees (degree=5,6,7,8,9) on the scatter plot so that we will have a better visualization of the accuracy:

*Polynomial Model (Degree: 6,7,8,9) on Scatter Plot*



Before we choose the best metric to evaluate our models, let’s go back and rethink what is our business goal. The data given only have one predictor: employee number per company. Response variable is the average cost in benefits. There are three possible scenarios:

* The managers will use this prediction model to help them decide better company size with the goal of minimizing the company cost on benefits to pass through the upcoming recession.
* Or the manager will use this model to decide how many employees should be fired or hired in the next season.
* Or the manager will use this model as a reference to adjust the cost on employee’s benefits. Are we overpaying benefits to our employee? If so, why? And how much of benefits in the next year?

Higher prediction accuracy is more important comparing to the deviation of the prediction results. Therefore, I choose to use root mean square error to check whether our predicted cost far off the target and also use R^2 to check how much our model performed better than the baseline (naïve mean model). Therefore the model minimizes error ( Low RMSE) and maximize prediction accuracy (High R^2) will be selected for the company.

|  |  |  |
| --- | --- | --- |
| Model | R\_2 | Mean Square Error |
| Random Forest | -29.1026 | 176.0 |
| Gradient Boosting | -30.123 | 180.1 |

I also tried to use random forest and gradient boost models. Both performed poorly comparing to polynomial. Maybe the data is truly coming from a linear model and that is why the step functions in tree-based regression performed poorly in our case.

PS:

One thing needs to be addressed: before using random forest or gradient boost, we need to transform the ‘cost’ variable from float to encode labels, such as integer:

We use preprocessing.LabelEncoder() package from Sklearn:

