# Answer Question 3A：

As we discuss in ‘Modeling\_Specification.docx’ document, we can use different ways to mitigate Overfitting issue in small dataset.

1) Introduce more data into dataset and reduce insignificant feature. But the amount of feature should be less than number of instances.

2) Resample the data structure to make features much balanced.

2) Decrease the complexity of model through adjusting parameters in models or adopting much simple model to fit.

3) Change the Penalty Function in Algorithm.

4) Introduce Cross-Validation method to keep modeling boosted.

To mitigate overfitting problem, we tried different models:

1)Linear Regression

Except Linear Regression model, we tried two models which introduced penalty parameter into linear regression: Lasso and Ridge regression. However, metrics show that the performance of model didn’t go robust. XGBoost model was a prevalent model using in prediction area of data science. However, even though the metrics shown that the performance of training and validation dataset was improved, the overfitting issue was exacerbated. In sum, to predict total revenue customer spent, we picked the relatively best model – Ridge model to fit the dataset and complete prediction. The main metrics we used to evaluate the models were Root Mean Squared Error (RMSE) and R-Square.

To fit the probability whether total revenue was beyond 250, we train our models by Logistic Regression which could predict the probability of 1-0 labeled dependent variables. We adopted Over-Sample and Under-Sample method to balance the 1-0 label and introduce model ensemble in order to avoid information loss. The metrics shown that Under-Sample and Over-Sample method indeed improved the performance of logistic regression and mitigate overfitting problem. In sum, we picked the best model – model ensemble using Under-Sample method to predict the probability of total revenue more than 250. The main metrics we used were confusion matrix and AUC score.

# Answer Question 3B:

To evaluate the accuracy of regression model, many metrics could be chosen to evaluate the performance of one model and compare the results of multiple models. The common metrics are MAE, RMSE, MSE, R-square and so on.

1) Mean Absolute Error (MAE):

MAE measures the mean of discrepancy between prediction and true values. The lower value MAE is, the better performance of model is. **The MAE is a basic metric to measure the accuracy of prediction.** MAE increases with variance of the errors.

2) Root Mean Squared Error (RMSE):

RMSE measures square root of average of squared discrepancy between prediction and true values. The lower value RMSE is, the better performance of model is. Because of squared calculation of discrepancy, RMSE will automatically assign high weight to large errors in prediction. As a result**, RMSE becomes more useful when large errors should be avoided in largest degree in modeling processes**. Moreover, RMSE increases with the variance of the frequency distribution of error magnitudes instead of variance of the errors themselves.

3) R-Square

R-Square is scaled between 0 and 1. It measures the ratio of squared discrepancy between prediction and mean of true value and variance of ture value multiplying by N. It doesn’t directly measure the accuracy of fit, instead, it considers the variance of original data and check whether this variance changes in prediction set. **So it only produces the ‘relative’ measure of error of modeling.** It is easy to interpret because it has upper and lower boundary. One problem is that when adding more predictors into model, R-Square will increase. To avoid this problem, Adjusted R-Square was introduced. Adjusted R-Square considers degree of freedom of model in order to decrease R-Square value when predictors increase.

To evaluate binary classification modeling, Classification Accuracy, Confusion Matrix and AUC score are mainly used to compare result of modeling.

1) Classification Accuracy

The most basic and common evaluation metric for classification problems. It measures the percentage of correct prediction verse total instances. However**, it is feasible on only condition that binary labeled variable is balanced.** If the dataset is extremely unbalanced, lets say only 2% is labeled as 0, Classification Accuracy will be 98% if model predicts all instances as 1, which is no usable.

2) Confusion Matrix

Confusion Matrix list all of binary label in dataset verse binary prediction in a matrix. It measures each amount of four categories: Ture Positive: ‘1’ is predicted to ‘1’, False Positive: ‘o’ is predicted to ‘1’, True Negative: ‘0’ is predicted to ‘0’, False Negative: ‘0’ is predicted to ‘1’. Based on those four numbers, **we could calculate the specific accuracy rate to make model result aligning with business sense**. However, it has no unique metric to evaluate the overall performance of classification model.

3) AUC Score and ROC Curve

ROC curve is created based on different cutoff of probability of ‘1’. The x axis is False Positive Rate, the y axis is True Positive Rate. They will be changed following increase of probability cutoff. The diagonal line in this curve presents the accuracy of random prediction of ‘1’ and ‘0’. AUC Score calculates the area under ROC curve which counts from 0 to 1. The diagonal line has 0.5 AUC. Normally, the AUC of prediction through modeling should range from 0.5 to 1, the bigger the better. **The AUC score grants analyst a good way to evaluate the performance of modeling**, but it is hard to connect this accuracy with business sense.

In specific project, it is hard to pick one metric over other possible metric because every metrics have their usability and limitation. So, to evaluate result of modeling more accurately, we can combine several metrics together. In linear regression, if we focus on reducing the overall errors in prediction, MAE or its deformations could be used. If large errors should be avoided specifically, RMSE could play well. Adjusted R-Square could be used to measure model that the predictor would not change and it is a great tool to interpret the fitting degree of modeling. In classification problem, overall classification Accuracy will work if the dataset was labeled in different categories equally. AUC score is use more commonly to evaluate the accurate of modeling. Confusion Matrix works well when applying to specific business problem. There are also lots of other metrics could be useful, such as Mean Absolute Percentage Errors (MAPE), GINI Score, Log-oriented metrics. Some of them may be executed commonly specific industry, such as GINI Score in Insurance industry. Some of them could be supplementary criteria to evaluate models.

In sum, the selection of metrics depends on specific problem and its business setting. To select model or improve performance of models in order to solve practical business issue, combination of several metrics but deep understanding about utility of them are essential.

Reference:  
1. <https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d>

2. <http://www.theanalysisfactor.com/assessing-the-fit-of-regression-models/>

3. <https://machinelearningmastery.com/metrics-evaluate-machine-learning-algorithms-python/>

# Answer Question 3C:

We can describe ways to mitigate the overfitting problem, if we have already got dataset large enough, in two aspects: dataset and modeling.

1) Dataset

If we face overfitting problem, first thing we should pay attention to is data exploratory analysis. Correlation between independent variables, significant of each variable in modeling if applicable and the quality of data resource should be considered. Data should be modified to perform regularization and remove insignificant independent variables. Dataset size could be an effect in overfitting problem, however whether we should import more data should be carefully evaluate. After specific threshold, the performance of modeling will not be improved. So, if we plan to import more data, research should be done to judge the necessity of this method.

2) Modeling

In modeling, we can adopt lots of way to deal with overfitting problem. First, we can evaluate the model based on other metrics. Evaluating model through multiple metrics could gain more accurate understanding of its performance. Second, we can reduce the ‘complexity’ of modeling through switching to much simple model, or changing parameters in modeling structure. Third, in data exploratory analysis, we have selected several features and get their significant degree to prediction. Some relatively insignificant feature could be deleted in order to mitigate the overfitting problem. To apply this method, we should do more investigations in feature engineering to ensure the features we got have almost extract information within the dataset. Fourth, we adjust levels of k-folder validation process to improve the accuracy of model response to diversity of instances.

Actually, if the dataset is large enough, instead of regression or classification, we can take advantage of deep learning technique to fit the dataset. We could build up multiple level of predictive modeling set and import specific features into certain levels. In that way, we can use simple model to avoid the overfitting effect but remain the learning depth to improve the accuracy of results. In deep learning net structure, we can reduce the features that import into perceptron and keep the depth of structure to the maximum extent to avoid overfitting. Regularization could be used to decease the complexity of modeling, such as dropout which means removing several perceptron randomly. However, Regularization may cause information loss in modeling.

In sum, there are lots of techniques and methods to deal with overfitting. When to sue them should depend on specific dataset and certain target problem largely. The fitting result would be most important criteria to judge which techniques work best.

Reference:

1. <http://www.holehouse.org/mlclass/10_Advice_for_applying_machine_learning.html>

2. <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>

3. <https://medium.com/towards-data-science/deep-learning-3-more-on-cnns-handling-overfitting-2bd5d99abe5d>