Test Task - 1 for Machine Learning [Stirring Minds]

Name: Chaitanya Jindal
Date: 3rd September 2023

Problem Statement

You are assigned to a project where you are working on a dataset provided with Career guidance which contains various details of the students and their preferred career choices. How do you select the important variables? Explain your method in detail. (You are free to assume the variables as needed).

Solution

Selecting essential variables in a career guidance dataset is crucial for building effective predictive models and extracting meaningful insights. Here is a step-by-step method to select essential variables:

1. Understand the Dataset:

- Begin by thoroughly examining the dataset's structure. What columns (variables) are present? What type of data do they contain (numeric, categorical, text, etc.)?
- Identify the target variable, which is likely to be the students' actual career choices. Understanding this variable is critical, as it will be the focus of your predictive modelling.

2. Data Exploration:

- Conduct Exploratory Data Analysis (EDA) to get a feel for the data.
- Visualize data distributions using histograms, box plots, and density plots to identify patterns.
- Compute summary statistics (mean, median, standard deviation) to understand central tendencies and variations.
- Create scatter plots or pair plots to explore relationships between variables.
- Detect any missing data or outliers, as these may impact the validity of your analysis.

3. Domain Knowledge:

- Consult with career guidance experts or relevant professionals who know the domain.
- They can provide valuable insights into which variables are likely to influence a student's career choice the most. For instance, factors like academic performance, extracurricular activities, and personal interests may play significant roles.

4. Correlation Analysis:

- Calculate correlations between variables, especially with the target variable (career choices).
- A correlation matrix or heatmap can visually represent these relationships.
- Variables with high positive or negative correlations with the target variable are candidates for further investigation.

5. Feature Importance Techniques:

- If you plan to use machine learning models, these models often have built-in methods for ranking feature importance.
- For instance, if you are using a Random Forest classifier, you can obtain feature importance after fitting the model.
- Consider using these techniques to identify which variables have the most impact on predicting career choices.

6. Recursive Feature Elimination (RFE):

- RFE is an iterative technique that starts with all features and removes the least important ones in each iteration.
- Train your predictive model at each step and evaluate its performance.
- Continue until you reach a predetermined number of features or notice that model performance stabilizes.

7. SelectKBest (for Classification):

- SelectKBest is a feature selection method based on statistical tests.
- For classification tasks, it evaluates the significance of each feature's relationship with the target variable using statistical tests like chi-squared or ANOVA.
- Select the top K features based on the test scores.

8. Regularization Techniques:

- In linear models like logistic regression, you can apply regularization techniques such as L1 (Lasso) or L2 (Ridge) regularization.
- These techniques add penalty terms to the model's coefficients, which encourages the model to select fewer but more important features.

9. Cross-Validation:

- Utilize cross-validation to assess how well your selected variables generalize to new data.
- This involves splitting your dataset into multiple subsets and training/validating your model on different combinations.
- Evaluate whether your model's performance remains consistent across these iterations.

10. Evaluate Model Performance:

- Continuously monitor your model's performance as you select variables.
- Use appropriate metrics for your specific task, such as accuracy for classification or RMSE for regression.
- Compare performance metrics for different variable subsets to identify the most effective set of variables.

11. Validate with Experts:

- Before finalizing your variable selection, validate your choices with domain experts and stakeholders.
- Ensure that the selected variables align with their knowledge and expectations about what influences career choices.

12. Documentation:

- Document your variable selection process thoroughly. Keep a record of the variables you selected and the reasons behind those choices.
- This documentation is essential for transparency, reproducibility, and sharing insights with others.

Test Task - 2 for Machine Learning [Stirring Minds]

Name: Chaitanya Jindal
Date: 3rd September 2023

Problem Statement

You were assigned to a project where you built a random forest model with 10,000 trees. You were on cloud nine after getting a training error of 0.00. But the validation error is 46.89. What went wrong? Does that mean that you trained your model wrong? (Explain briefly) And how would you explain Machine learning to a grade-1 kid (probably 5-6 years old)?

Solution

1st Part:

A training error of 0.00 and a validation error of 46.89 indicate a classic case of overfitting. In this scenario, it means that the random forest model with 10,000 trees has learned the training data almost perfectly, achieving a training error of zero. However, it failed to generalize well to new, unseen data, which is reflected in the high validation error.

Overfitting occurs when a model learns the noise and specific patterns in the training data, rather than capturing the underlying patterns that can be applied to new data. In this case, having an excessively large number of trees (10,000) in the random forest may have contributed to the overfitting. Random forests are robust against overfitting, but extremely deep and complex forests can still overfit if not properly controlled.

To address this issue and improve the model's performance:

1. **Reduce the Number of Trees:**
   - Consider reducing the number of trees in the random forest. Often, a smaller number of trees can generalize better.
2. **Tune Hyperparameters:**
   - Experiment with hyperparameters such as the maximum depth of trees, the minimum number of samples required to split a node, or the maximum number of features considered for splitting. This can help control the complexity of individual trees in the forest.
3. **Feature Engineering:**
   - Revisit the feature selection and engineering process. Sometimes, overfitting can be mitigated by selecting more relevant features or transforming existing ones.
4. **Cross-Validation:**

- Use cross-validation to better estimate the model's performance on unseen data during training. It can help detect overfitting early and fine-tune the model accordingly.

5. **Regularization:**

- Random forests do not typically require regularization, but if overfitting remains an issue, you can explore techniques like feature bagging or reducing the depth of individual trees.

Remember that machine learning is an iterative process, and it is common to encounter overfitting. The key is to strike a balance between model complexity and generalization by fine-tuning hyperparameters and optimizing the model until it performs well on both the training and validation datasets.

2nd Part:

Machine learning is like teaching a computer to learn things, just like how you learn from your experiences and mistakes. Imagine you have a pet robot, and you want it to recognize different kinds of fruits, like apples and bananas.

First, you show your robot lots of pictures of apples and tell it, "This is an apple." Then, you show it many pictures of bananas and say, "This is a banana." The robot looks at these pictures and tries to understand what makes an apple different from a banana.

Now, when you show your robot a new picture of a fruit, it will try to guess if it is an apple or a banana based on what it learned from the pictures you showed it before. If it guesses right, that is great! If it is wrong, you can tell your robot that it made a mistake, and it will try to learn from that and do better next time.

So, machine learning is like teaching computers to be smart and make decisions by showing them lots of examples and helping them get better at it over time, just like you learn and get better at things as you grow up.