Lab3 Answers

1. We retained the following features:
   1. Asn\_peer\_participants
   2. Proto\_unicast\_mgmt\_publics
   3. Public\_id\_mgmt\_public\_ips
   4. Proto\_multicast\_mgmt\_publics
   5. Info\_traffic\_peer\_participants
   6. Info\_ratio\_peer\_participants
   7. Info\_scope\_peer\_participants
   8. Info\_prefixes\_peer\_participants
   9. Policy\_general\_peer\_participants
   10. Policy\_locations\_peer\_participants
   11. Policy\_ratio\_peer\_participants
   12. Policy\_contracts\_peer\_participants
   13. Info\_type\_peer\_participants
   14. Proto\_ipv6\_mgmt\_publics

These features seem to have more discriminatory power than the others. Many of the features have nearly-unique values for each AS (there is very little variance) – these likely won’t help our classifier. We looked for features that were standardized (common input formats), with moderate variance

1. There are 139 cases remaining after removing unimportant features and removing cases with blank (NaN) entries in the important features.

|  |  |
| --- | --- |
| Variance Threshold | Removed Features |
| 0.9 | Policy\_ratio\_peer\_participants  Proto\_unicast\_mgmt\_publics |
| 0.75 | Policy\_ratio\_peer\_participants  Proto\_unicast\_mgmt\_publics  Proto\_multicast\_mgmt\_publics |
| 0.5 | proto\_ipv6\_mgmt\_publics  proto\_multicast\_mgmt\_publics  policy\_ratio\_peer\_participants  policy\_contracts\_peer\_participants  proto\_unicast\_mgmt\_publics |

1. Rank of feature importance according to tree-based estimator:

info\_ratio\_peer\_participants 0.17627299

asn\_peer\_participants 0.14835012

info\_prefixes\_peer\_participants 0.13891334

public\_id\_mgmt\_publics\_ips 0.12507067

info\_traffic\_peer\_participants 0.1074613

info\_scope\_peer\_participants 0.096029

policy\_locations\_peer\_participants 0.05029597

policy\_general\_peer\_participants 0.04891598

proto\_ipv6\_mgmt\_publics 0.0451145

proto\_multicast\_mgmt\_publics 0.02995568

policy\_contracts\_peer\_participants 0.020589

policy\_ratio\_peer\_participants 0.01303143

proto\_unicast\_mgmt\_publics 0

We were surprised that info\_ratio and asn are the two most important features according to the tree-based estimator. Especially asn. This means the ASN itself, is a good indicator of the buissness type.

1. The tree-based estimator removed the following features. Based on its ranking of the features (question 13), this makes sense.

proto\_ipv6\_mgmt\_publics

policy\_locations\_peer\_participants

proto\_multicast\_mgmt\_publics

proto\_unicast\_mgmt\_publics

policy\_general\_peer\_participants

policy\_ratio\_peer\_participants

policy\_contracts\_peer\_participants

1. GNB accuracy w/ variance-based feature selection: 0.229

GNB precision w/ variance-based feature selection: 0.167

GNB recall w/ variance-based feature selection: 0.229

GNB accuracy w/ tree-based feature selection: 0.342

GNB precision w/ tree-based feature selection: 0.520

GNB recall w/ tree-based feature selection: 0.342

1. Decision tree accuracy w/ variance-based feature selection: 0.571

Decision tree precision w/ variance-based feature selection: 0.601

Decision tree recall w/ variance-based feature selection: 0.571

Decision tree accuracy w/ tree-based feature selection: 0.343

Decision tree precision w/ tree-based feature selection: 0.383

Decision tree recall w/ tree-based feature selection: 0.343

1. Random forest accuracy w/ variance-based feature selection: 0.514

Random forest precision w/ variance-based feature selection: 0.537

Random forest recall w/ variance-based feature selection: 0.514

Random forest accuracy w/ tree-based feature selection: 0.371

Random forest precision w/ tree-based feature selection: 0.472

Random forest recall w/ tree-based feature selection: 0.371

3 Feature selection

A critical part of data preparation and machine learning is feature selection. As you should have observed, our data is full of extraneous detail as it relates to business type classification.

1. Based on your intuition, which features do you think are valuable (features we want to keep) for a model in classifying an ASes business type? Explain why. A good place to start is by identifying the feature columns that are free-form text and non-standard (mixed data type) inputs.

“info\_traffic” is certainly a good feature to use for AS classification. Write a python program to standardize these values for each AS and remove any ASes that do not provide this data.

Before continuing, remove the features you identified in question 10. Write a python program to remove cases with blank entries. Lastly, write a python program that (for all features) maps each unique feature value to an integer. For example, for “info\_ratio\_peer\_participants”, Balanced = 0, Mostly Inbound = 1, Mostly Outbound = 2.

1. After removing the unimportant features and the cases with blank entries in your remaining features, how many cases are in the data? In your estimation, are there enough cases to achieve our objective? You will work with this version of the data for the remainder of the lab. Explain your answer.

Feature selection and dimensionality reduction can be done in a more scientific manner using variance, recursive methods, and decision trees (for example). Check out scikit-learn’s feature selection module, sklearn.feature\_selection (<https://scikit-learn.org/stable/modules/feature_selection.html>).

1. Eliminate all features from the data whose variance is lower than 0.9. Try it with a variance threshold of 0.75 and 0.5. Which features were removed for each variance threshold? Explain your results.
2. Tree-based estimators can also be used to compute feature importance and remove unnecessary features from a dataset. Using a tree-based estimator, rank the importance of the data features from most important to least important. Explain your results.
3. Use a tree-based estimator to remove unnecessary features from the data. Which features were removed. Explain your results.

Take some time to experiment with feature selection and elimination. Create a personalized version of our data with at least four important features.

4 Machine learning

Let’s see if how well we can predict an AS type based on the features we have selected. First, you should normalize your data matrix. This means looking at the range of values for a given column and scaling the values of that column such that they lie between [0, 1]. Also, shuffle your data before proceeding.

1. Using scikit-learn, write a program that uses Gaussian Naïve Bayes (GNB) to build a model. Split the data and train on 80% and test on 20%. What accuracy, precision, and recall do you achieve with GNB and using variance-based feature selection? With a tree-based estimator?
2. Using scikit-learn, write a program that uses a decision tree to build a model. Again, split the data and train on 80% and test on 20%. What accuracy, precision, and recall do you achieve with the decision tree and using variance-based feature selection? With a tree-based estimator?
3. Using scikit-learn, write a program that uses a random forest to build a model. Again, split the data and train on 80% and test on 20%. What accuracy, precision, and recall do you achieve with the random forest and using variance-based feature selection? With a tree-based estimator?