Practice Midterm - Applied Machine Learning COMS W4995

Date:

Name:

UNI:

1 True/False (+ 2pt each)

	True	False
A fast-forward git merge creates a new commit.		x
np.random.uniform() == np.random.uniform() evaluates to True.		x
Regression models based on minimizing mean squared error are sensitive to outliers.	x	
The sign of a particular coefficients in ridge regression will be the same, no matter what the regularization parameter.		x
Stochastic gradient descent is suitable for datasets with a very high number of samples.	x	
It is good practice to standardize sparse dataset so that each feature has zero mean.		x
A node in a decision tree always contains exactly half the samples of its parent.		x
Kernel support vector machines don’t scale well to large datasets.	x	
Decision Trees are very sensitive to the scaling of the data.		x
For a perfectly calibrated classifier, 80% of the data for which p(y=1) =.8 belong to class 1.	x	

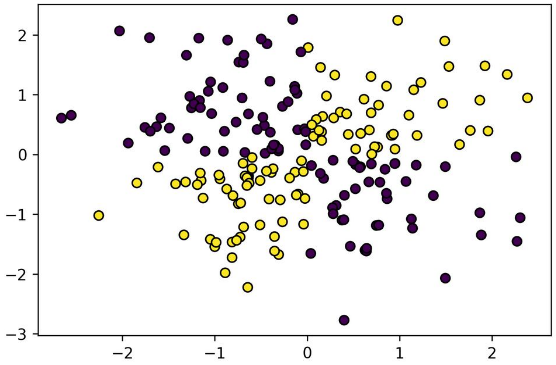
2 Multiple choice (20pt)

Select all choices that apply.

2.1 Which of the following are non-parametric models?

- ■ Random Forest
- ❏ Linear Regression
- ❏ Logistic Regression
- ■ Nearest Neighbors
- ❏ Nearest Shrunken Centroid

2.2 Given a two-class classification dataset with the two features shown below and additional non-informative features, which of the following feature selection methods would be able to identify the these two features as informative?



- ❏ SelectPercentile(f_classif)
- ❏ SelectKBest(mutual_info_classif)
- ■ SelectFromModel(DecisionTreeClassifier())
- ■ SequentialFeatureSelector(SVC(kernel=’rbf’))
- ❏ RFE(LogisticRegression())

2.3 Which of the following variables should be encoded as categorical?

- ❏ Income
- ■ Nationality
- ■ Gender
- ❏ Age
- ■ ZIP code

2.4 Which of the following transformations allow linear classifiers to learn non-linear decision boundaries?

- ❏ RobustScaler()
- ■ PolynomialFeatures(degree=2)
- ■ RBFSampler()
- ❏ SelectFromModel(DecisionTreeClassifier())

3 Debugging (10pt each)

For each code snippet, find and explain all errors given the task.

3.1 Task: Use cross-validation to assess how well feature selection and Random forest will do on the test set.

```
select = SelectPercentile(percentile=50).fit(X_train, y_train)
X_train_selected = select.transform(X_train)
scores = cross_val_score(RandomForestClassifier(n_estimators=100),
                         X_train_selected, y_train, cv=10)
```

Information leakage by using feature selection outside of cross-validation

3.2 Task: Use the PowerTransformer (implementing the box-cox transformation) transformer to preprocess data and learn a Ridge model, and visualize the coefficients.

```
pipe = make_pipeline(StandardScaler(), PowerTransformer(), Ridge())
scores = cross_val_score(pipe, X_train, y_train, n_folds=10)
plt.barh(range(X_train.shape[0]), pipe.coef_)
```

PowerTransformer can’t work on the negative data created by StandardScaler(), N_folds should be cv, pipe.coef_ doesn’t exit.

4 Coding (10 each)

Provide code to build a `LogisticRegression` model and evaluate its performance on a separate test set, given a classification dataset as numpy arrays X and y.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y)
lr = LogisticRegression().fit(X_train, y_train)
score = lr.score(X_test, y_test
```

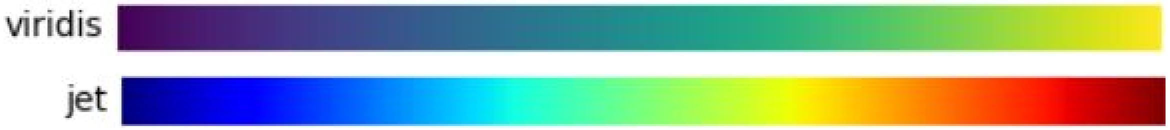
Provide code to implement grid-searching the parameters C and gamma of an SVC in a pipeline with a StandardScaler, and evaluating the best parameter setting on a separate test set, given data as numpy arrays X and y.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y)
Param_grid = {‘svc__C’: np.logspace(-3, 3, 7), ‘svc__gamma’: np.logspace(-3, 3, 7)}
# you could also use n_features to change the range of gamma. Either is acceptable.
pipe = make_pipeline(StandardScaler(), SVC())
grid = GridSearchCV(pipe, param_grid=param_grid, cv=5)
# cv not necessary, specifying a different scoring would also be fine
grid.fit(X_train, y_train)
score = grid.score(X_test, y_test)
```

5 Concepts (5pt each)

Answer each question with a short (2-5 sentences) explanation.

5.1 How are the “jet” and “viridis” colormaps different and why does it matter?



5.2 Explain the difference between Logistic regression and linear SVMs.

5.3 Explain the basic idea of the RANSAC algorithm.

5.4 Explain out-of-bag estimates of generalization error in random forests.