Final Exam - Applied Machine Learning COMS W4995

Date:

Name:

UNI:

1 True/False (+2 for correct/ - 2pt for incorrect, +/- 0 for unanswered.)

	True	False
The AUC score is independent of the decision threshold applied to the probability predicted by a classifier.	x	
The macro-average f1 puts more weight on rare classes than the micro-average f1.	x	
You can always extract as many principle components as there are input features.		x
With an imbalanced dataset, downsampling the majority class will lead to slower training than upsampling the minority class.		x
ARI is a practical way of adjusting the number of clusters in K-Means for exploratory data analysis.		x
A Gaussian Mixture Model allows evaluating the probability of a new point under a fitted model.	x	
The NMI is not defined for cluster assignments with different numbers of clusters.		x
Isolation Forests assume Gaussian Distributed Data		x
In a bag-of-word model with unigrams, using stop-words will drastically reduce the number of features.		x
Backpropagation is an algorithm to optimize the weights of a neural network.		x

2 Multiple choice (20pt)

Select all choices that apply.

2.1 2.1 Which of the following statements apply to neural networks?

- ❏ Fast to train.
- ✓ Can learn arbitrarily complex functions.
- ❏ Work well when little training data is available.
- ✓ Provide state-of-the-art performance in computer vision and audio analysis.

2.2 Which of the following models are generative probabilistic models of the data?

- ❏ NMF
- ✓ Latent Dirichlet Allocation
- ✓ Gaussian Mixture Models
- ✓ PCA
- ❏ t-SNE
- ❏ KMeans

2.3 What are reasons to prefer NMF over PCA?

- ❏ Better reconstruction of the data.
- ✓ Sign of the components is meaningful.
- ✓ No cancellation effects.
- ✓ Can extract non-linear features.
- ❏ Faster.
- ❏ Deterministic results.

2.4 Which of the following cluster evaluation methods are unsupervised?

- ✓ Silhouette Score
- ❏ ARI
- ❏ NMI
- ✓ Stability based score

3 Debugging (10pt each)

For each code snippet, find and explain all errors given the task. Assume all necessary imports have been made. There can be more than one error per task!

3.1 Task: Perform grid-search on a Keras Sequential model for the number of units (50, 100 or 200) in the hidden layer. The network should be a one-hidden-layer network for 64 input features and 8 classes.

```
X_train, X_test, y_train, y_test = train_test_split(X, y)
model = Sequential([Dense(50),
                    Dense(8, activation=”softmax”))

model.compile(“adam”, “multiclass_crossentropy”,
metrics=[“accuracy”])

param_grid = {‘hidden_units’: [50, 100, 200]}
grid = GridSearchCV(model, param_grid)
grid.fit(X_train, y_train)
score = grid.score(X_test, y_test)
```

No input shape defined.
No non-linear activation function specified.
Can’t use “sequential” in a GridSearchCV, need to define a callable and give it to KerasClassifir.

3.2 Task: Write down the computation in a forward-pass of a feed-forward neural network for classification with one hidden layer with 100 units, sigmoid non-linearity (logistic sigmoid 1/1 +exp(-x) given as `sigm`) and a drop-out rate of 50% on the hidden layer.

```
def forward(X, w1, b1, w2, b2):
    h1_net = np.dot(X, w1 + b1)
    dropout_mask = np.random.uniform(100) > .5
    h1_net[dropout_mask] = .5
    h1 = sigm(h1_net)
    out_net = np.dot(X, w2 + b2)
    out_exp = np.exp(out_net)
    return out_exp - np.sum(out_exp)
```

Bias should be added after matrix multiplication.
Drop-out needs to happen after non-linearity.
Need to divide by np.sum(out_exp) for softmax.

4 Coding (10 each)

Assume all necessary imports have been made.

4.1 Define a multi-layer perceptron with relu non-linearity and a single hidden layer with 100 hidden units for classifying the iris dataset.

Sequential([Dense(100, input_shape=(4,), activation="relu"),
Dense(3, activation="softmax")])

4.2 Implement training an EasyEnsemble for classification with Decision Trees on X_train, y_train and prediction on X_test, y_test

```
probs = []
for i in range(n_estimators):
    est = make_pipe(RandomUnderSampler(),
DecisionTreeRegressor(random_state=i))
    est.fit(X_train, y_train)
    probs.append(est.predict_probab(X_test, y_test))
pred = np.argmax(np.mean(probs, axis=0), axis=1)
# using VotingClassifier as on the slides would also be fine
# optionally set max_features="sqrt"
```

5 Concepts (5pt each)

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

5.1 Explain the “CBOW” approach used in word2vec. How are the word representations found?
Given a word in one-hot encoding, try to predict all words in the surrounding context. Prediction is done using two matrix multiplications (like a linear neural net), where the hidden layer corresponds to the size of the embedding vectors. The prediction is done using softmax. The model is learned using SGD sampling words and contexts from the training data.

5.2 Explain how “batch normalization” works.
During the forward pass in a neural network, before the non-linearity, the activations are normalized to have zero mean and unit variance. This is done separately for each mini-batch, and re-computed for each iteration. An additional scaling factor and offset is learned on the standardized activations to maintain the same representational power. The gradient computation takes the normalization into account. Normalizing activations in this way leads to faster learning.

5.3 Explain the Rand Index (without adjustment for chance).
For two partitionings C1 and C2, the Rand Index counts how many of all possible pairs of samples are in the same subset in C1 that are also in the same subset in C2. This count is added to the count of the number of pairs of samples that are in different subsets in C1 and are also in different subsets in C2.
This sum is normalized by the number of all pairs of samples, n choose 2.

5.4 Give 3 reasons why convolutional neural networks are better suited for image recognition than fully connected networks.
Possible reasons:
They can exploit the neighborhood structure of images.
They are invariant to translations.
They require learning less parameters.
The can share parameters between different locations in the image.
They can be used for feature extraction on differently-sized images.

6. Bonus question (there won't be one in the exam)!
What TV shows have been referenced in the slides and homeworks of this course?

- ❏ Firefly
- ✓ Archer
- ❏ Death Note
- ✓ Rick and Morty
- ❏ Steven Universe
- ✓ One Punch Man