AWS Course - The Elements of Data Science - part 4

Lesson 10 of 12

Model Evaluation and Model Productionizing

Productizing a ML Model

aws training and certification

Aspects to consider:

- Model hosting
- Model deployment
- Pipelines to provide feature vectors
- Code to provide low-latency and/or high-volume predictions
- Model and data updating and versioning
- Quality monitoring and alarming
- Data and model security and encryption
- Customer privacy, fairness, and trust
- Data provider contractual constraints (e.g., attribution, cross-fertilization)

Types of production environments

aws training and certification

Batch predictions

- Useful if all possible inputs known *a priori* (e.g., all product categories for which demand is to be forecast, all keywords to bid)
- Predictions can still be served real-time, simply read from pre-computed values

Online predictions

- Useful if input space is large (e.g., customer's utterances or photos, detail pages to be translated)
- Low latency requirement (e.g., at most 100ms)

Online training

- Sometimes training data patterns change often, so need to train online (e.g., fraud detection)

Model Evaluation Metrics

Business Metrics vs. Model Metrics

aws training and certification

- Business metrics may not be the same as the performance metrics that are optimized during training. Why?
  - Example: Click-through rate
- Ideally, performance metrics are highly correlated with business metrics.

Confusion Matrix

aws training and certification

Predicted class

Actual class	P	N
P	True Positives (TP)	False Negatives (FN)
N	False Positives (FP)	True Negatives (TN)

Breast Cancer Example

True Label \ Predict Label	1 (M)	0 (B)
1 (M)	146	24
0 (B)	11	274

*Data Source: Breast Cancer Wisconsin (Diagnostic) Data Set

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Also called *score*

Precision

aws training and certification

Issue: In many applications, TN dwarfs the other categories, making accuracy useless for comparing models.

Precision: Proportion of positive predictions that are actually correct

- $Precision = \frac{TP}{TP+FP}$
- Example: $\frac{146}{146+11} = 0.9299$

Recall

aws training and certification

Issue: In many applications, TN dwarfs the other categories, making accuracy useless for comparing models.

Recall: Proportion of positive set that are identified as positive

- $Recall = \frac{TP}{TP+FN}$
- Example: $\frac{146}{146+24} = 0.8588$

F1-Score

Issue: In many applications, TN dwarfs the other categories, making accuracy useless for comparing models.

F_1-Score: Combination (harmonic mean) of precision and recall

- $F_1 - Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$

Cross-Validation

This video explains cross-validation, a technique used to assess how the results of one model will generalize against a new data set.

Cross-Validation

Issue

Metrics on training data can't measure generalization.

- Model could cheat by memorizing the data and getting a perfect score
- Overfitting

Solution

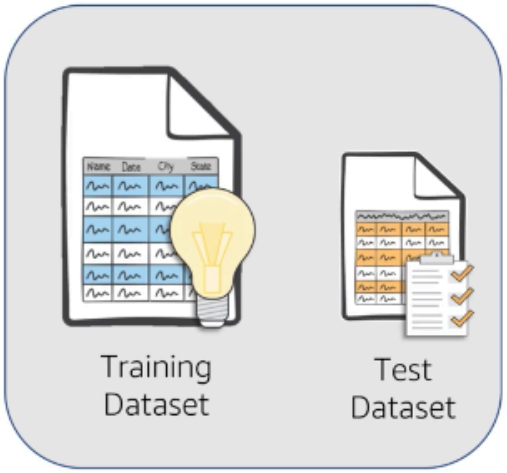
- Cross-validation: Train and evaluate on distinct data sets.

Holdout Method

aws training and certification

Split data set into separate training and test sets:

- Training set: Used to train, tune, and select model
- Test set: Used to evaluate final model



Using Training and Testing Data Sets

aws training and certification

```
from sklearn.model_selection import train_test_split

# split data into training and test sets
train, test = train_test_split(df[col + ['target']],
                               test_size = 0.3)

# Fit the SVM model
clf = svm.SVC()
clf.fit(train[col], train['target'])
```

```
# Check the confusion matrix and score for training data
pred = clf.predict(train[col])
print(confusion_matrix(y_true=train['target'],
                       y_pred=pred, labels=[1, 0]))

print(clf.score(train[col],train['target']))
```

```
# Check the confusion matrix and score for test data
pred = clf.predict(test[col])
print(confusion_matrix(y_true=test['target'],
                       y_pred=pred, labels=[1, 0]))

print(clf.score(test[col], test['target']))
```

Still Overfitting—Too Many Variables

aws training and certification

*Data Source: Breast Cancer Wisconsin (Diagnostic) Data Set

Training Dataset

Predict Label

True Label	Confusion Matrix	1 (B)	0 (M)
	1 (B)	258	0
	0 (M)	2	138

Score = 0.9949

Testing Dataset

Predict Label

True Label	Confusion Matrix	1 (B)	0 (M)
	1 (B)	98	1
	0 (M)	66	6

Score = 0.6081

K-Fold Cross-Validation

aws training and certification

- **Issue: Small sets**
  - Smaller training set → not enough data for good training
  - Unrepresentative test set → invalid metrics
- **K-Fold cross-validation**
  - Randomly partition data into K "folds"
  - For each fold, train model on other K-1 folds and evaluate on that
  - Train on all data
  - Average metric across K folds estimates test metric for trained model

K-Fold CV: Choosing K

aws training and certification

- Large → more time, more variance

- Small → more bias

- 5-10 value of K is typical

K-Fold Cross-Validation

K-Fold Cross-Validation: Special Cases

Leave-one-out cross-validation

- K = number of data points
- Used for very small sets

Stratified K-fold cross-validation

- Preserve class proportions in the folds
- Used for imbalanced data
- There are seasonality or subgroups

Metrics for Linear Regression

Mean Squared Error

aws training and certification

- Average squared error over entire dataset

$$Mean\ squared\ error(MSE) = \frac{1}{N}\sum_{i=1}^{N}\left(\widehat{y_i} - y_i\right)^2$$

- Very commonly used
- scikit-learn: sklearn.metrics.mean_squared_error

R^2: Coefficient of Determination

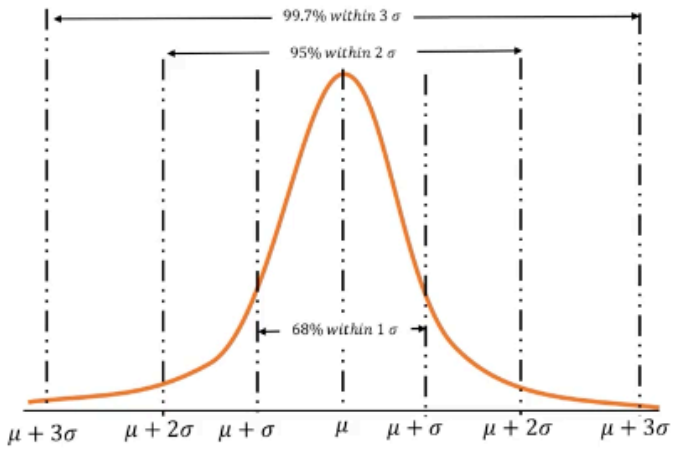
aws training and certification

- $R^2 = 1 - \frac{Sum\ of\ Squared\ Error\ (SSE)}{Var(y)}$ which is between 0 and 1
- Interpretation: Fraction of variance accounted for by the model
- Basically, standardized version of MSE
- Good R^2 are determined by actual problem
- R^2 always increases when more variables are added to the model

$$\text{Adjusted } R^2 = 1 - (1 - R^2)\frac{no.\ of\ data\ pts. - 1}{no.\ of\ data\ pts. - no.\ of\ variables - 1}$$

- Takes into account of the effect of adding more variables such that it only increases when the added variables have significant effect in prediction

Normal (Gaussian) Distribution



Probability Density Function

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

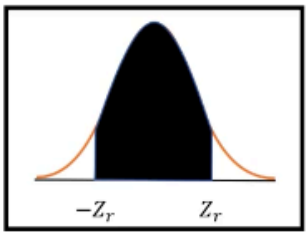
- μ is mean or expectation of the distribution
- σ is standard deviation
- σ^2 is variance

Why do we study normal distribution so often?

Central Limit Theorem: no matter what is the original distribution of X, the mean of X (i.e. $\bar{X}$) will follow a normal distribution: $\bar{X} \sim N\left(\mu, \frac{\sigma^2}{n}\right)$.

Confidence Interval

- An average computed on a sample is merely an *estimate* of the true population mean.
- **Confidence interval**: Quantifies margin-of-error between sample metric and true metric due to *sampling randomness.*
- **Informal interpretation**: With x% confidence, true metric lies within the interval.
- **Precisely**: If the true distribution is as stated, then with x% probability the observed value is in the interval.
- **Z-score**: Quantifies how much the value is above or below the mean in terms of its standard deviation



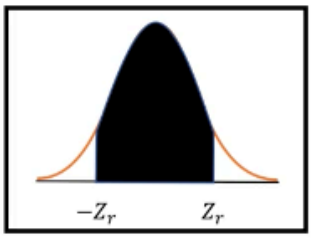
Prob = r	z-score = z_r
0.90	1.645
0.95	1.96
0.98	2.326
0.99	2.576

Confidence Interval

aws training and certification

For population proportion (i.e. truth), the confidence interval is:

- CI = $p \pm z(p(1 - p)/n)^{1/2}$
- Where p is sample proportion, n is sample size n, and z is z-score defined in the table above, which is determined by the confidence level.



Prob = r	z-score = z_r
0.90	1.645
0.95	1.96
0.98	2.326
0.99	2.576

Using ML Models in Production: Storage

Data Storage Formats

aws training and certification

- Row-oriented formats:
  - Comma/tab-separated values (CSV/TSV)
  - **Read-only DB (RODB)**: Internal read-only file-based store with fast key-based access
  - **Avro**: allows schema evolution for Hadoop
- Column-oriented formats:
  - **Parquet**: Type-aware and indexed for Hadoop
  - **Optimized row columnar (ORC)**: Type-aware, indexed, and with statistics for Hadoop
- User-defined formats:
  - **JavaScript object notation (JSON)**: For key-value objects
  - **Hierarchical data format 5 (HDF5)**: Flexible data model with chunks
- Compression can be applied to all formats
- Usual trade-offs: Read/write speeds, size, platform-dependency, ability for schema to evolve, schema/data separability, type richness

Model and Pipeline Persistence

aws training and certification

Predictive Model Markup Language (PMML):

- Vendor-independent XML-based language for storing ML models
- Support varies in different libraries:
  - **KNIME (analytics/ML library)**: Full support
  - **Scikit-learn**: Extensive support
  - **Spark MLlib**: Limited support

Custom methods:

- **Scikit-learn**: Uses the Python pickle method to serialize/deserialize Python objects
- **Spark MLlib**: Transformers and Estimators implement MLWritable
- **TensorFlow (deep learning library)**: Allows saving of MetaGraph
- **MxNet (deep learning library)**: Saves into JSON

Model Deployment

aws training and certification

Technology transfer: Experimental framework may not suffice for production

- A/B testing or shadow testing: Helps catch production issues early

Information Security

aws training and certification

- Make sure that you handle training and evaluation data in accordance with data classification.

Using ML Models in Production: Monitoring and Maintenance

Monitoring

- It's important to monitor quality metrics and business impacts with dashboards, alarms, user feedback, etc.:
  - The real-world domain may change over time.
  - The software environment may change.
  - High profile special cases may fail.
  - There may be a change in business goals.

Maintenance

- Performance deterioration may require new tuning:
  - Changing goals may require new metrics.
  - A changing domain may require changes to validation set.
  - Your validation set may be replaced over time to avoid overfitting.

Customer Obsession

- Think carefully about impact on customer perception and trust.
- Give ML solutions the "creepiness sniff test" and "*The* front page of a newspaper test."
- Provide explanations to customers.

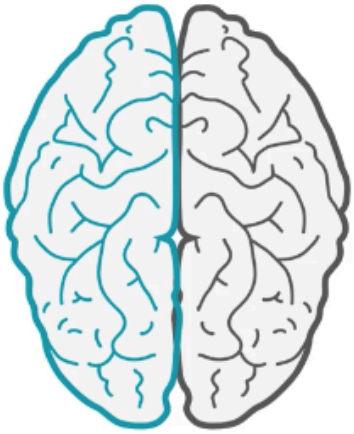
Using ML Models in Production: Using AWS

Here, we explore AWS's comprehensive ecosystem for building machine learning models.

Artificial Intelligence on AWS

aws training and certification

Powerful **tools** for all developers



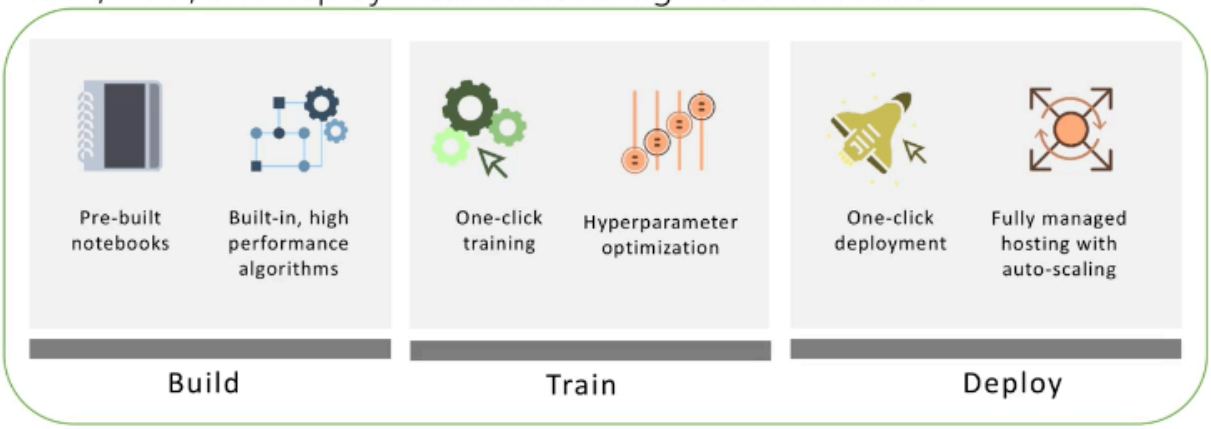
to add **intelligence** to your applications

Artificial Intelligence on AWS

aws training and certification

Amazon SageMaker

Build, train, and deploy machine learning models at scale.



Artificial Intelligence on AWS

aws training and certification

Amazon Rekognition Image

Deep learning-based **image** analysis

Amazon Rekognition Video

Deep learning-based **video** analysis

Built on technology used by
Amazon Prime Photos
to analyze billions of images daily

Artificial Intelligence on AWS

aws training and certification

Amazon Lex

Build chatbots to engage customers

Uses the same technology as
Amazon Alexa
to provide advanced
deep learning functionalities
Of ASR and NLU

Artificial Intelligence on AWS

aws training and certification

Amazon Polly

Natural sounding text to speech, which includes:

- More than two dozen languages
- Wide variety of natural-sounding voices

Artificial Intelligence on AWS

aws training and certification

Amazon Comprehend

Natural language processing (NLP) service that enables you to:

- Discover insights and relationships in text
- Identify language based on the text
- Extract key phrases, places, people, brands, or events
- Understand how positive or negative the text is
- Automatically organizes a collection of text files by topic

Artificial Intelligence on AWS

aws training and certification

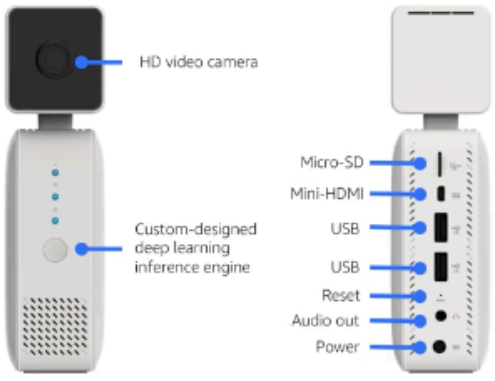
Amazon Translate

Neural machine translation service that enables you to:

- Perform fluent translation of text
- Localize content for international users
- Easily translate large volumes of text efficiently

AWS DeepLens

aws training and certification



HD video camera with on-board compute optimized for deep learning

Integrates with Amazon SageMaker and AWS Lambda

10 MIN

From unboxing to first inference in <10 minutes

Tutorials, examples, demos, and pre-built models

More ML Frameworks and Services

- **AWS Glue:** Data integration service for managing ETL jobs
- **Deep Scalable Sparse Tensor Network Engine (DSSTNE):** Neural network engine

Common Mistakes

This sections review common mistakes that occur during machine learning projects.

Common Mistakes in ML Projects

- You solved the **wrong problem.**
- The data was **flawed.**
- The solution didn't **scale.**
- Final result **doesn't match** with the prototype's results.
- It takes **too long to fail.**
- The solution was **too complicated.**
- There weren't enough allocated engineering **resources** to try out long-term science ideas.
- There was a lack of a true **collaboration.**

Knowledge Check 5

Here's what you said:

Here's what we said:

In classification problems, we usually set the interested responses as positive class and in many cases, the interested class is related to a rare situation (for example fraudulent transaction) and the regular class are more common (for example regular transaction). TN cases are usually dominate and if you include TN and focus on accuracy, then it may give us very high accuracy, but not much detail about the model performance for the positive class (i.e. relative composition of TP, FP and FN). For a balanced data where positive and negative cases are roughly equal, we can include TN in the analysis.

Q2

Write the equation for accuracy in terms of the confusion matrix.

- **A.** Accuracy = TP/TN
- **B.** Accuracy = (TP+TN)/(TP+FP+FN+TN)
- **C.** Accuracy = TP+TN
- **D.** Accuracy = TP/(FN+FP)

Which ONE of the metrics is appropriate for measuring the quality of a *regression* solution:

a) R^2
b) Recall
c) F1 score
d) False positive rate

Q4

What is precision, and recall for the following classification prediction results:

		Prediction	
		True	False
Actual	True	30	12
	False	8	64

- (●) **A.** Precision: 0.789, Recall: 0.714
- ( ) **B.** Precision: 0.592, Recall: 0.968
- ( ) **C.** Precision: 0.728, Recall: 0.602
- ( ) **D.** Precision: 0.808, Recall: 0.882