Machine Learning Bootcamp
IDUG NA 2019: June 6, 2019

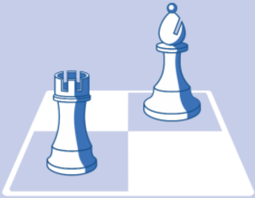
Mark Ryan mryan@ca.ibm.com
Jason Shayer shayer@ca.ibm.com
Steve Darling darling@ca.ibm.com
Roger Sanders roger.sanders@us.ibm.com

What is Machine Learning?

Artificial Intelligence (AI)

Human intelligence exhibited by machines

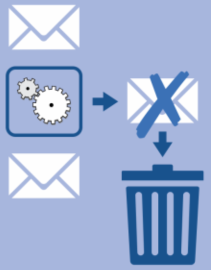
- Reasoning
- Natural Language Processing (NLP)
- Planning



Machine Learning (ML)

An approach to achieve AI

- Gradient Boosting Machine (GBM)
- Support Vector Machine (SVM)
- Logistic Regression
- Factorization Machines (FM)
- Field-aware Factorization Machines (FFM)

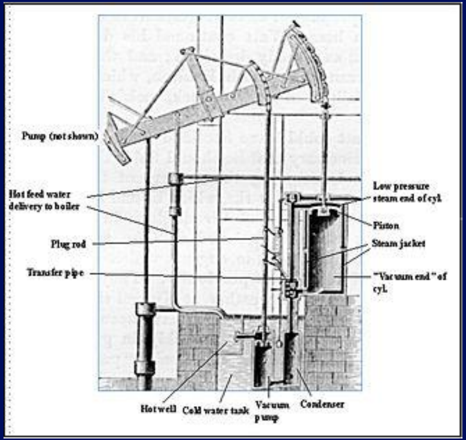


Deep Learning (DL)

A technique for implementing ML

- Deep Neural Networks
- Deep Belief Networks
- Recurrent Neural Networks

Where are we in the lifecycle of Machine Learning?



1770's:
- Watt's stationary engine
- Capital-intensive one-off applications

ML era: late 2000s

1829:
- Stephenson's Rocket
- Standardization and regular service

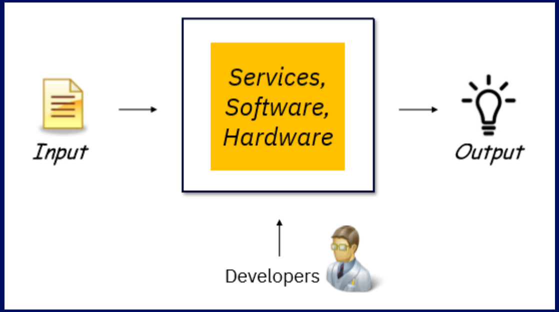
ML era: 2019

1941:
- Alleghany locomotive
- Apex of steam technology

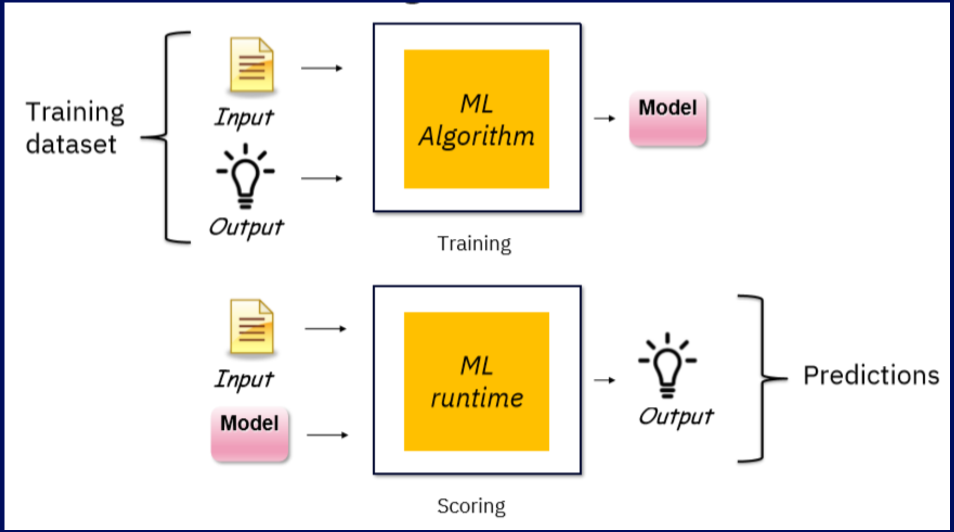
ML era: 5-10 years from now?

What is Machine Learning?

Classical Programming



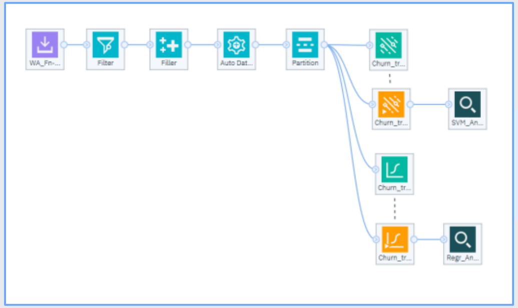
Machine Learning



Customer churn: use case to illustrate machine learning

- **Customer churn**: when a customer ends their relationship with a business
- **PROBLEM**: You need to predict which of your customers are loyal and which are at risk of churning
- **SOLUTION**: Use data you have about clients to build a model that can predict whether a given client is going to churn

Customer Churn: the game plan



1

Solve the customer churn problem using Modeler

2

Solve the customer churn problem using Python

Python: Ingest Dataset

Click on the cell below to highlight it.

Then go to the `Files` section to the right of this notebook and click `Insert to code` for the data you have uploaded. Choose `Insert pandas DataFrame`.

```
url="https://raw.githubusercontent.com/ryanmark1867/june2019_ML_bootcamp/master/WA_Fn-UseC_-Telco-Customer-Churn.csv"
customer_data = pd.read_csv(url)
customer_data.head()
```

:[6]:

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	No	No	No	Month-to-month	Yes	Electronic check	29.85
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	No	No	No	One year	No	Mailed check	56.95
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	No	No	No	Month-to-month	Yes	Mailed check	53.85
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes	No	No	One year	No	Bank transfer (automatic)	42.30
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	No	No	No	Month-to-month	Yes	Electronic check	70.70

5 rows × 21 columns

```
The model predicts that this client will churn
```

	MonthlyCharges	TotalCharges	InternetService	PaymentMethod	OnlineSecurity	Contract	tenure
0	90.5	1791.5	Fiber optic	Credit card (automatic)	No	Month-to-month	20.0

3

Define a pipeline in Python to predict if a client will churn

Customer churn: dataset

- CSV (comma separated values) file with ~7k records; 21 columns
- Numeric columns: **tenure, MonthlyCharges**, **TotalCharges**
- Categorical columns: **gender**, **SeniorCitizen**, **Partner**…
- Target / Label: **Churn**

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Yes	No	DSL	Yes	No	No	No

Customer churn: preparing the dataset for ML

Original dataset

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Yes	No	DSL	Yes	No	No	No

Why?

Keep a subset of columns — *Pick just the columns needed for the model*

Deal with missing values — *ML algorithms need numeric input*

Deal with outliers — *Control impact of anomalies*

Scale continuous values — *Consistent impact of columns*

Deal with string values — *ML algorithms need numeric input*

Split Dataset into Train & Test — *Need to reserve a portion of the data the model has never seen to validate the model*

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Yes	No	DSL	Yes	No	No	No

Customer churn: preparing the dataset for ML

Original dataset

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Yes	No	DSL	Yes	No	No	No

Keep a subset of columns

Deal with missing values

Deal with outliers

Scale continuous values

Deal with string values

Split Dataset into Train & Test

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Yes	No	DSL	Yes	No	No	No

Pick the 8 columns used for the model:

- ***tenure***
- ***InternetService***
- ***OnlineSecurity***
- ***Contract***
- ***PaymentMethod***
- ***MonthlyCharges***
- ***TotalCharges***
- ***Churn***

Customer churn: the pipeline

Original dataset

customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBac	DevicePro	TechSup
7590-VHVEG	Female	0	Yes	No	1	No	No phone serv	DSL	No	Yes	No	No
5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes	No
3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No	No
7795-CFOCW	Male	0	No	No	45	No	No phone serv	DSL	Yes	No	Yes	Yes
9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No	No
9305-CDSKC	Female	0	No	No	8	Yes	Yes	Fiber optic	No	No	Yes	No
1452-KIOVK	Male	0	No	Yes	22	Yes	Yes	Fiber optic	No	Yes	No	No
6713-OKOMC	Female	0	No	No	10	No	No phone serv	DSL	Yes	No	No	No
7892-POOKP	Female	0	Yes	No	28	Yes	Yes	Fiber optic	No	No	Yes	Yes
6388-TABGU	Male	0	No	Yes	62	Yes	No	DSL	Yes	Yes	No	No
9763-GRSKD	Male	0	Yes	Yes	13	Ye	No	DSL	Yes	No	No	No

Split Dataset into Train & Test

Keep a subset of columns

Deal with missing values

Deal with outliers

Scale continuous values

Deal with string values

Logistic Regression model

Pipeline:

- ***Train the data preparation steps and the model in one operation***
- ***Apply pipeline to get a churn / no churn prediction for a given client***
  - ***Performs data prep on client's data***
  - ***Applies model to get a prediction***

Machine learning models: Logistic Regression

- Classification: churn / no churn
- Extension of **linear regression**
  - simplest algorithm; used to predict continuous values
  - e.g. predict house price from # of bedrooms, sq. ft, frontage
- Logistic regression is “geared” to output between 0 and 1
  - treat 0.5 as the boundary

Machine learning models: Logistic Regression

- Define function: **$\hat{Y}$ = h = sigmoid(Xθ)**
  - **$\hat{Y}$** is the prediction (*predicted churn values*)
  - **X** is the input
  - **θ** is an array of weights

Sigmoid(z) = $\frac{1}{1+e^{-z}}$

Not quite!

X

MonthlyCharges	TotalCharges	InternetService	PaymentMethod	OnlineSecurity	Contract	tenure
29.85	29.85	DSL	Electronic check	No	Month-to-month	1
56.95	1889.50	DSL	Mailed check	Yes	One year	34
53.85	108.15	DSL	Mailed check	Yes	Month-to-month	2
42.30	1840.75	DSL	Bank transfer (automatic)	Yes	One year	45
70.70	151.65	Fiber optic	Electronic check	No	Month-to-month	2

Y

Churn
No
No
Yes
No
Yes

Machine learning models: the secret sauce

- Define a **loss function** (delta between predictions **Ŷ** and actual values **Y**):

$$h = g(X\theta)$$
$$J(\theta) = \frac{1}{m} \cdot \left(-y^T \log(h) - (1-y)^T \log(1-h)\right)$$

- Repeatedly update **θ** (weights):

  1. Calculate the *partial derivative* of the loss function with respect to the weights = **X(Ŷ – Y)**
  2. Update the weights by subtracting the partial derivative

- What does making updates to the weights based on “the slope” of the loss function do?
- **With these repeated updates to the weights, the loss function gets minimized and the accuracy of the model gets maximized**

Machine learning models: *all values must be numeric!*

X

Y

MonthlyCharges	TotalCharges	InternetService	PaymentMethod	OnlineSecurity	Contract	tenure
29.85	29.85	DSL	Electronic check	No	Month-to-month	1
56.95	1889.50	DSL	Mailed check	Yes	One year	34
53.85	108.15	DSL	Mailed check	Yes	Month-to-month	2
42.30	1840.75	DSL	Bank transfer (automatic)	Yes	One year	45
70.70	151.65	Fiber optic	Electronic check	No	Month-to-month	2

Churn
No
No
Yes
No
Yes

data preparation

MonthlyCharges	TotalCharges	InternetService	PaymentMethod	OnlineSecurity	Contract	tenure
-1.160323	-0.992611	0	2	0	0	-1.277445
-0.259629	-0.172165	0	3	2	1	0.066327
-0.362660	-0.958066	0	3	2	0	-1.236724
-0.746535	-0.193672	0	0	2	1	0.514251
0.197365	-0.938874	1	2	0	0	-1.236724

Churn
0
0
1
0
1

Machine learning models: Logistic Regression

- Define function: **Ŷ = h = sigmoid(Xθ)**
  - **Ŷ** is the prediction (*predicted churn values*)
  - **X** is the input
  - **θ** is an array of weights

Sigmoid(z) = $\frac{1}{1+e^{-z}}$

Yes!

X

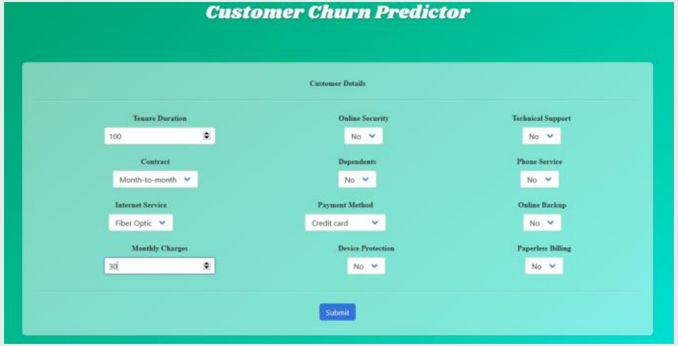
MonthlyCharges	TotalCharges	InternetService	PaymentMethod	OnlineSecurity	Contract	tenure
-1.160323	-0.992611	0	2	0	0	-1.277445
-0.259629	-0.172165	0	3	2	1	0.066327
-0.362660	-0.958066	0	3	2	0	-1.236724
-0.746535	-0.193672	0	0	2	1	0.514251
0.197365	-0.938874	1	2	0	0	-1.236724

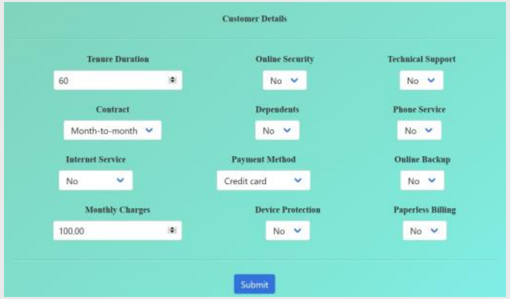
Y

Churn
0
0
1
0
1

Customer churn: exercising the model

- churn_match_modeler-scoring.ipynb to exercise the model in a notebook
- full-blown Python project to deploy and exercise the model:





Machine learning: more background

- Overall:
  - Andrew Ng machine learning intro course: https://www.coursera.org/learn/machine-learning
  - Fast.ai deep learning course: https://course.fast.ai/
- Details:
  - Sklearn pipelines: https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html
  - Logistic Regression math & implementation in Python: https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html
  - Articles on a variety of machine learning topics: https://medium.com/@markryan_69718