MACHINE LEARNING WITH ML.NET AND AZURE DATA LAKE

ANDY CROSS

Director Elastacloud / Azure MVP / Microsoft Regional Director

Many thanks to our sponsors & partners!

PLATINUM

GOLD

SILVER

PARTNERS

POWERED BY

Who am I?

- Andy Cross; andy@elastacloud.com; @andyelastacloud

- Microsoft Regional Director
- Azure MVP for 7 years

- Co-Founder of Elastacloud, an international Microsoft Gold Partner specialising in data

What am I here to talk about?

- Machine Learning
- A new dotnet approach to data
  - ML.NET (https://dotnet.microsoft.com/apps/machinelearning-ai/ml-dotnet)
  - Parquet.NET
- Big Data in production
  - Relevant tools in Azure for data

And what does it mean to say “Artificial Intelligence”

WHAT IS MACHINE LEARNING?

Machine learning for prediction talk quality

Formal definition: -

> "A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*."

E = historical quantities of beer consumed and corresponding audience ratings

T = predicting talk quality

P = accuracy of the talk quality prediction

Examples today

- Sentiment analysis of website comments
- Finding mappings of historic taxi fares
- Predicting the most likely group of flowers a certain flower belongs to
- Time Series
- Image Recognition

- NOTE: We reuse algorithms; to do so in most cases is akin to writing a new database to write a website – a vast overreach of engineering

Machine learning best practice

Machine learning should be approached in a methodical manner, in this way we are more likely to achieve accurate, reliable and generalisable models

This is achieved by following best practice for machine learning

Best practice mostly revolves around how the data is used

Machine learning best practice

Data preparation

- Ensure that the feature do not contain future data (aka time-travelling)
- Training, validation and testing data sets

Cross validation

- Think of this as using mini training and testing data sets to find a model that generalises to the problem

Validation and testing data sets

- Data that the model has never seen before – simulate the future
- Gives a final ‘sanity’ check of our model’s performance

Model training

Provide an algorithm with the training data set

The algorithm is ‘tuned’ to find the best parameters that minimise some measure of error

Model testing

To ensure that our model has not overfit to the training data it is imperative that we use it to predict values from unseen data

This is how we ensure that our model is generalisable and will provide reliable predictions on future data

Use a validation set and test sets

Validation set

The validation set is randomly chosen data from the same data that the model is trained with – but not used for training

Used to check that the trained model gives predictions that are representative of all data

Used to prevent overfitting

Gives a 'best' accuracy

Test set

The test set data should be ‘future’ data

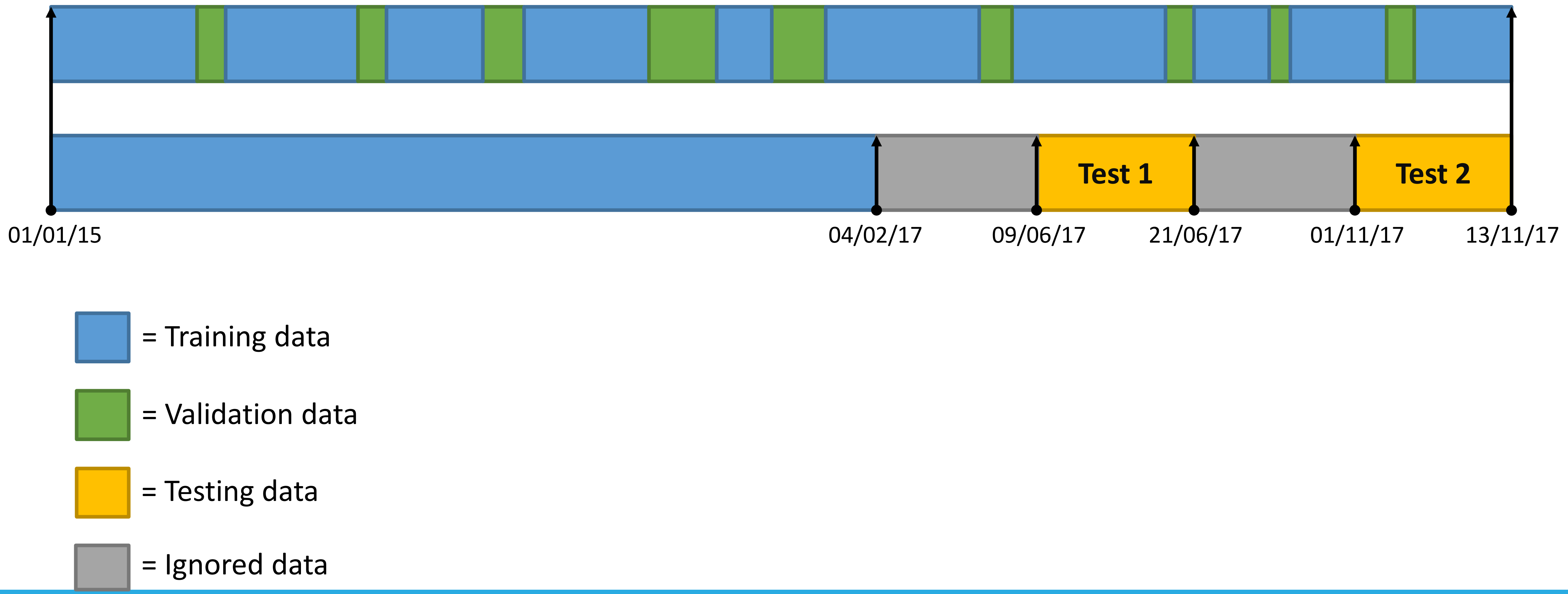
We simulate this by selecting data from the end of the available time-frame with a gap

Use the trained model to predict for this

More realistic of the model in production

Gives a conservative estimate of the accuracy

Validation and test sets



Model training, validation and testing

Train with cross validation to find best parameters

Assess overfitting on validation set

Retrain with best parameters on training data set

Evaluate performance on test data sets

HOW DOES THIS RELATE TO BIG DATA?

Distributed Computing

- Break a larger problem into smaller tasks
- Distribute tasks around multiple computation hosts
- Aggregate results into an overall result

- Types of Distributed Compute
  - HPC – Compute Bound
  - Big Data – IO Bound

- Big Data – Database free at scale IO over flat files

Freed of historic constraints

- Algorithms such as we'll see tonight are not new
  - From the 1960s many
- The difference is the ability to show the algorithm more examples
- Removing IO bounds gives us access to more data
- Removing CPU bounds allows us to compute over larger domains

Azure Services for Distributed Compute

- Azure Batch
- Azure HDInsight
- Azure Databricks

- Bring your own partitioner
  - Azure Functions
  - Azure Service Fabric (Mesh)
  - Virtual Machines
  - ACS/AKS

AZURE SERVICES FOR ML

Various places for ML

- Azure Databricks
- Azure HDInsight
- Azure Data Lake Analytics
- Azure ML
  - V1
  - V2 with Workbench
  - V2 RC no Workbench
- R/Python in many hosts
  - Functions
  - Batch
  - SQL Database

- C# and dotnet hosted in many places

- Typical Azure DevOps pipelines more mature for .NET

Azure Databricks

- Databricks Spark hosted as SaaS on Azure
- Focussed on collaboration between data scientists and data engineers
- Powered by Apache Spark
- Dev in:
  - Scala
  - Python
  - Spark-SQL
  - More?

Collaboration in Databricks

Collaborative Editing

Switch between Scala, Python, R & SQL

On notebook comments

Link notebooks with GitHub

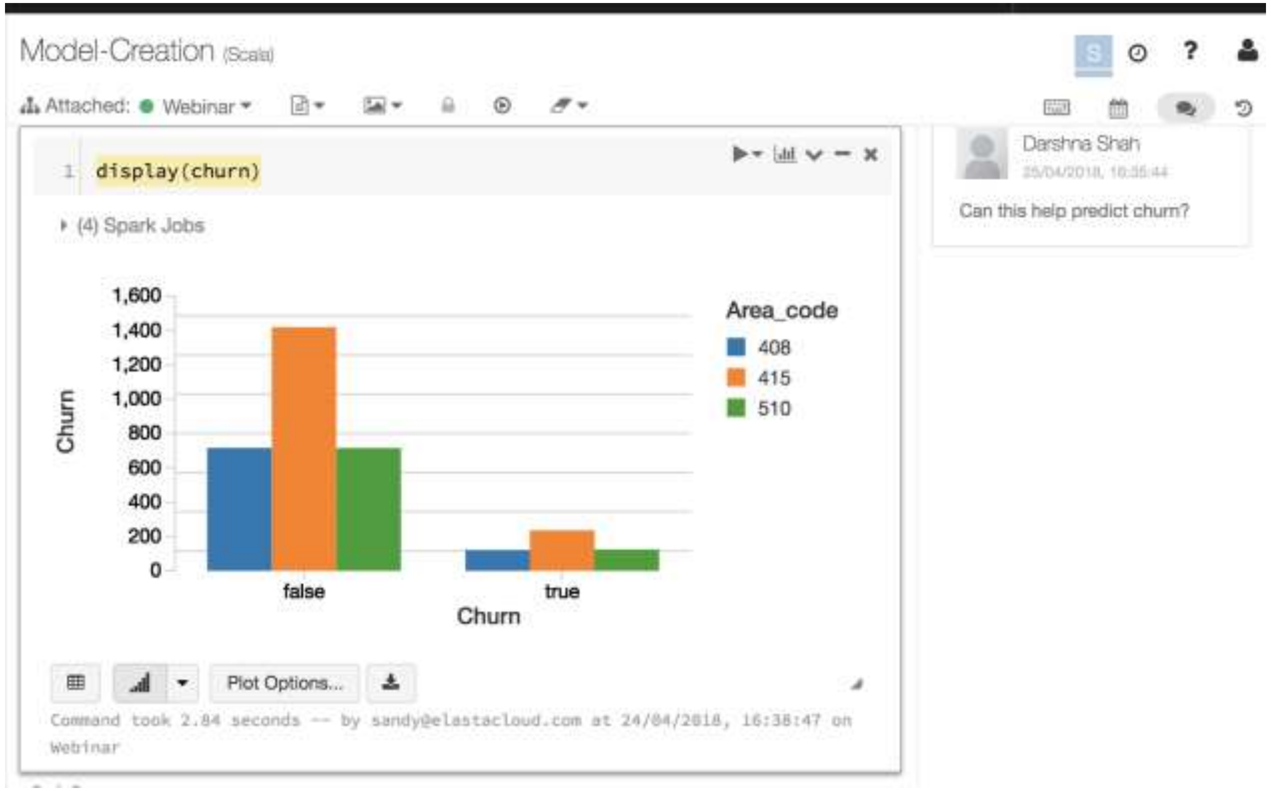
Databricks Notebooks Visuals

Visualisations made easy

Use popular libraries – ggplot, matplotlib

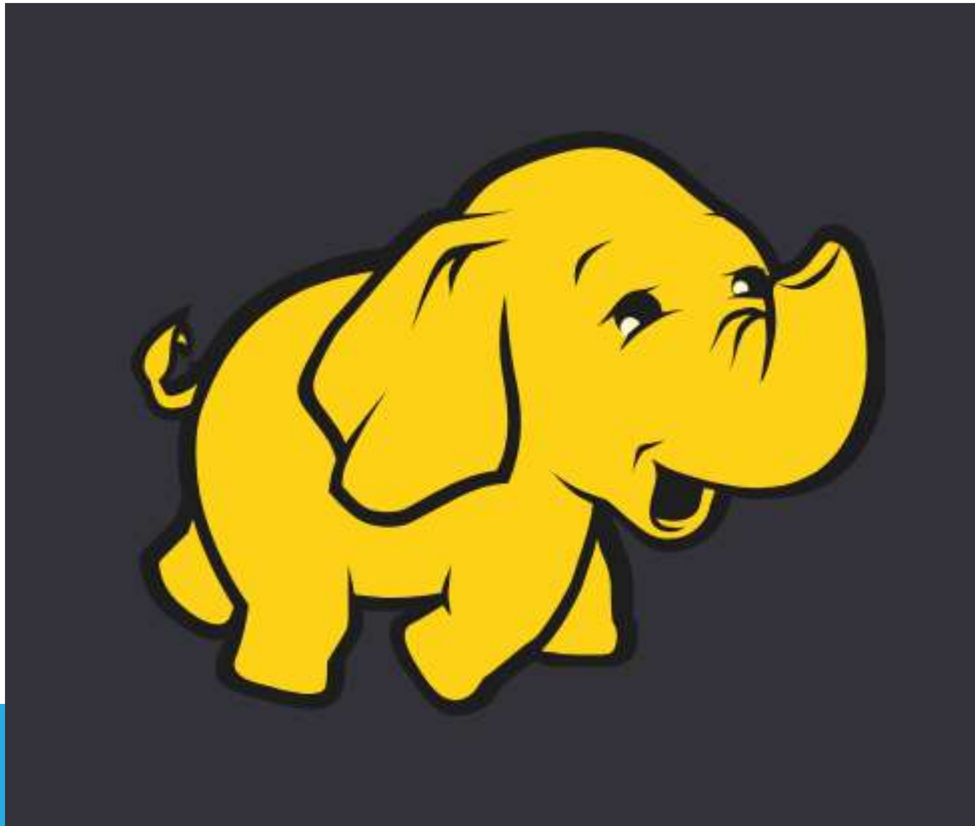
Create a dashboard of pinned visuals

Use Pip, CRAN & JAR packages to add additional visual tools

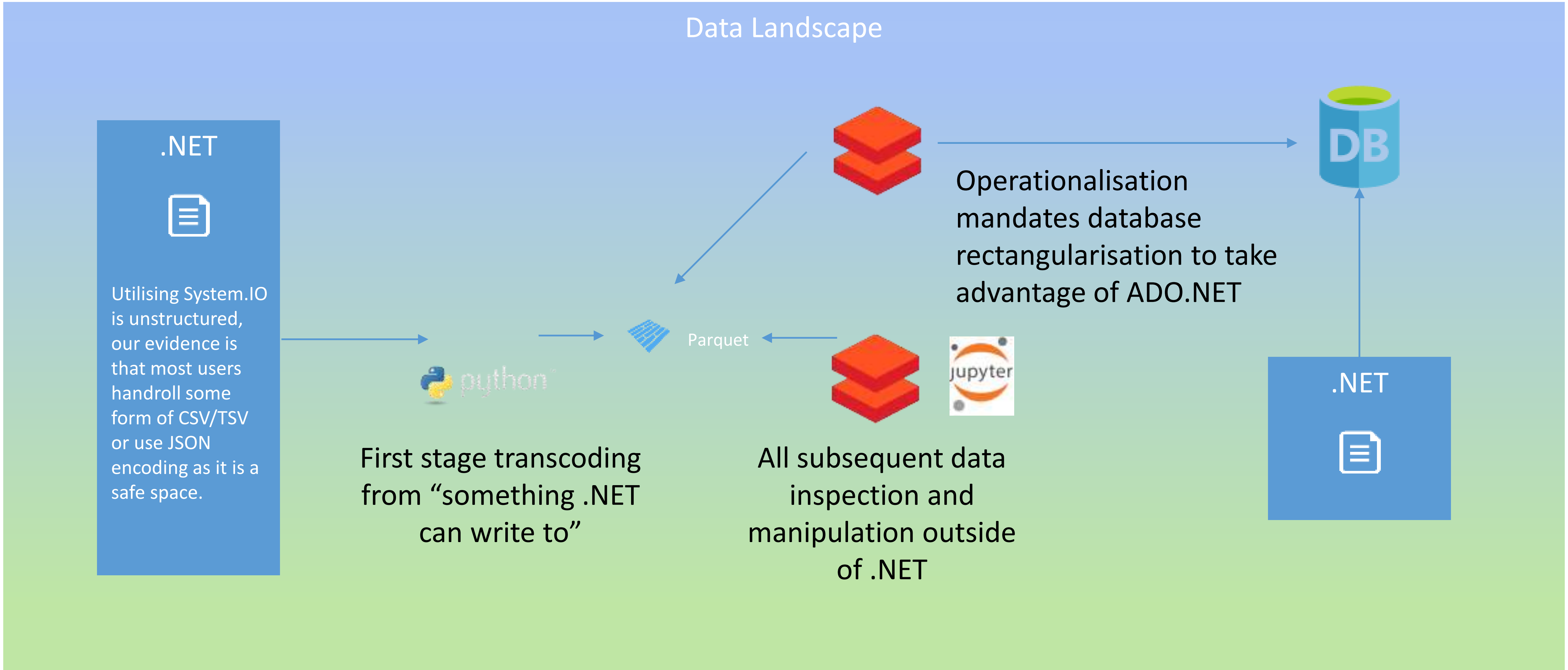


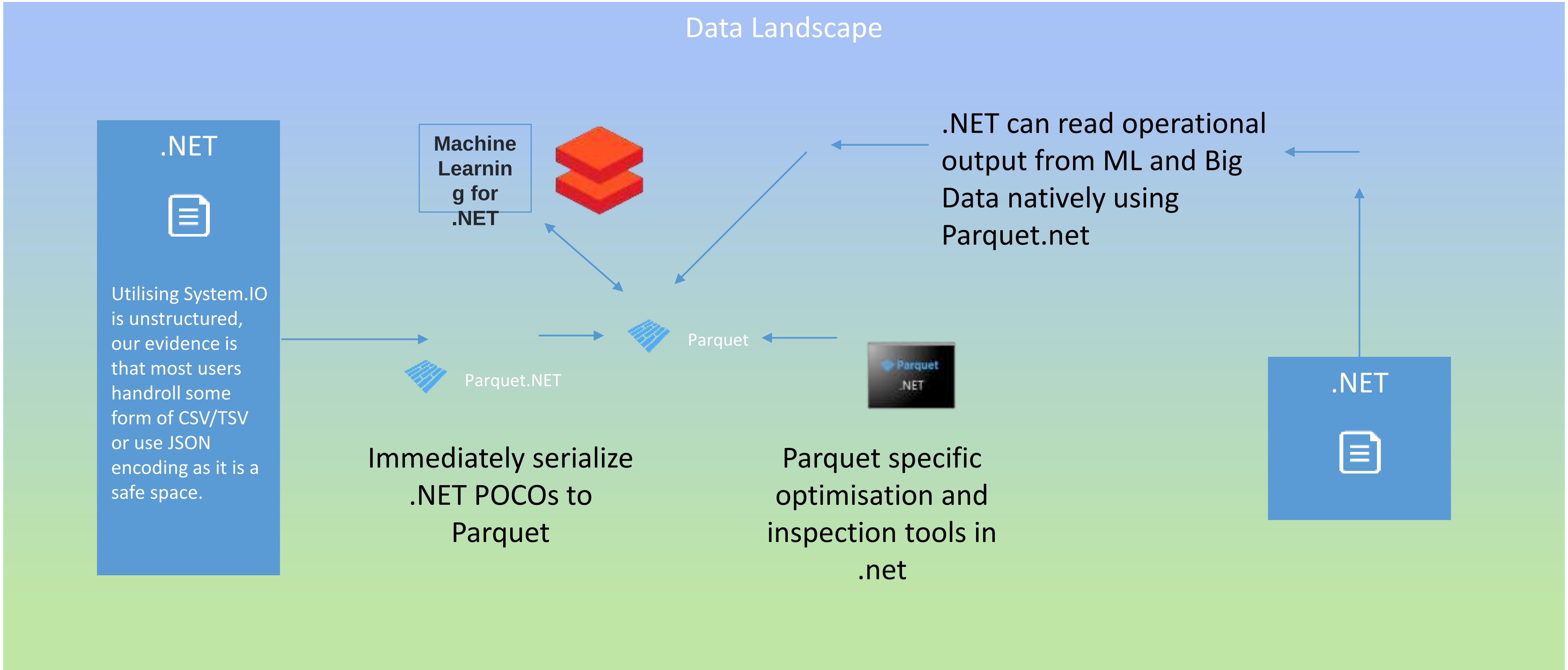
Azure HDInsight

- Multipurpose Big Data platform
- Operates as a host and configurator for various tools
  - Hadoop
  - Hive
  - Hbase
  - Kafka
  - Spark
  - Storm



REVOLUTIONISING DATA FOR DOTNET


Data Landscape
.NET
Utilising System.IO is unstructured, our evidence is that most users handroll some form of CSV/TSV or use JSON encoding as it is a safe space.
python
Parquet
First stage transcoding from “something .NET can write to”
All subsequent data inspection and manipulation outside of .NET
jupyter
Operationalisation mandates database rectangularisation to take advantage of ADO.NET
DB
.NET


Data Landscape
.NET
Utilising System.IO is unstructured, our evidence is that most users handroll some form of CSV/TSV or use JSON encoding as it is a safe space.
Machine Learnin g for .NET
Parquet
Parquet.NET
Parquet .NET
Immediately serialize .NET POCOs to Parquet
Parquet specific optimisation and inspection tools in .net
.NET can read operational output from ML and Big Data natively using Parquet.net
.NET

With Example in Sentiment Analysis with Binary Classifier

INTRODUCTION TO ML.NET

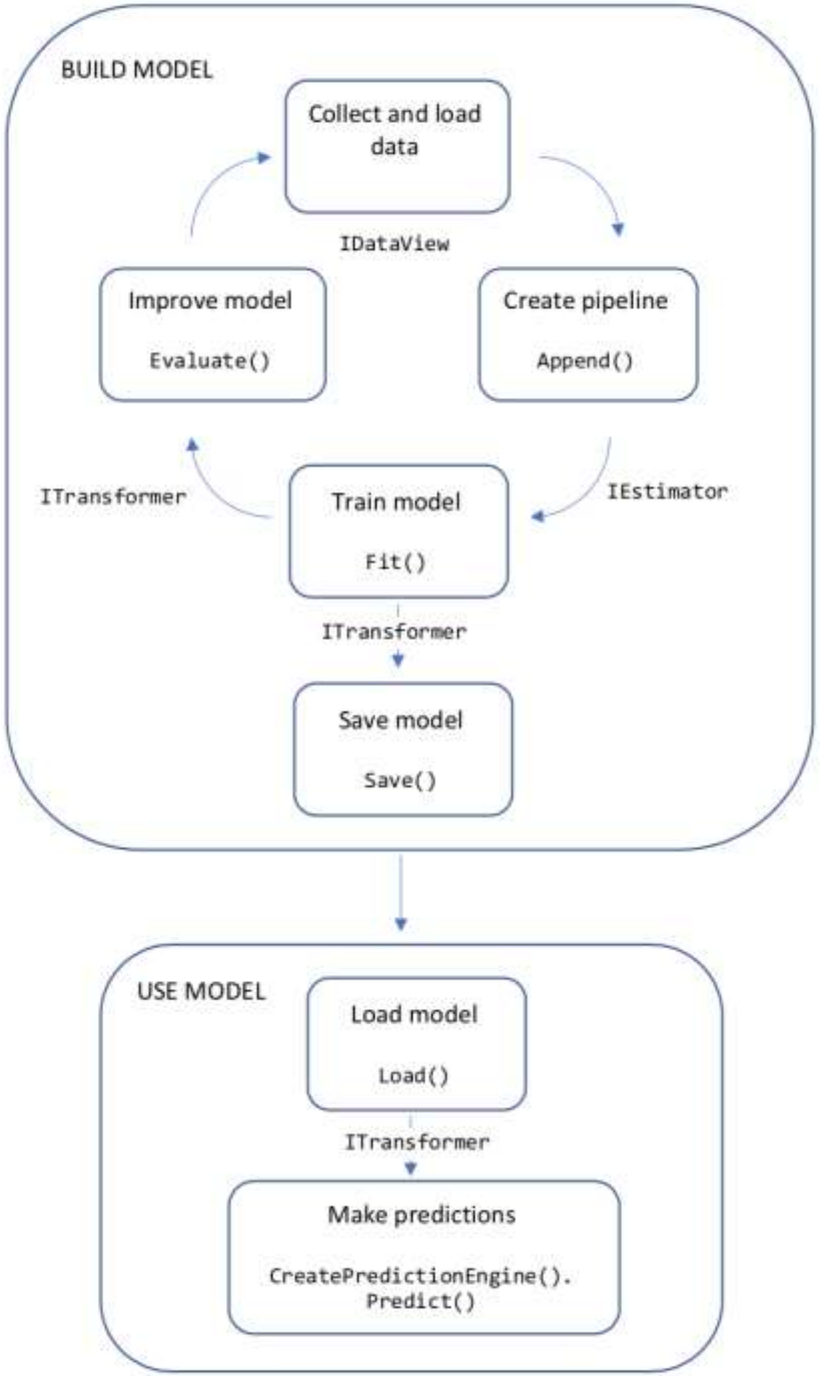
V1 Stable!!! ML.NET Pipelines

- Library evolving quickly
- Towards common approach in Spark/SciKit-Learn

- LearningPipeline weaknesses:
  – Enumerable of actions not common elsewhere
  – Logging and State not handled

- Bring in MlContext, Environment and lower level data tools IDataView

- Not all <0.6.0 features available, so later demos are "Legacy"


BUILD MODEL
Collect and load data
IDataView
Create pipeline
Append()
IEstimator
Train model
Fit()
ITransformer
Improve model
Evaluate()
ITransformer
Save model
Save()
USE MODEL
Load model
Load()
ITransformer
Make predictions
CreatePredictionEngine().
Predict()


Windows PowerShell
PS C:\>

Sentiment Analysis

- By showing examples of content and a Toxic bit flag, learn that makes things Toxic

- The quality of the model reflects the quality of the data
  - Representation
  - Variety
  - Breadth

Testing and Training Data

Tab Separated, two columned data set with headers.

Sentiment	SentimentText
1	==You're cool== You seem like a really cool guy... *bursts out laughing at sarcasm*.
0	I just want to point something out (and I'm in no way a supporter of the strange old git), but he is referred to as Dear Leader, and his father was referred to as Great Leader.
1	==RUDE== Dude, you are rude upload that carl picture back, or else.
0	" : I know you listed your English as on the ""level 2"", but don't worry, you seem to be doing nicely otherwise, judging by the same page - so don't be taken aback. I just wanted to know if you were aware of what you wrote, and think it's an interesting case. : I would write that sentence simply as ""Theoretically I am an altruist, but only by word, not by my actions."". : PS. You can reply to me on this same page, as I have it on my watchlist. "

```
private static IDataView GetData(LocalEnvironment env, string dataPath)
{
    var reader = new TextLoader(env,
                    new TextLoader.Arguments()
                    {
                        Separator = "tab",
                        HasHeader = true,
                        Column = new[]
                        {
                            new TextLoader.Column("Label", DataKind.Bool, 0),
                            new TextLoader.Column("Text", DataKind.Text, 1)
                        }
                    });

        //Load training data
    IDataView trainingDataView = reader.Read(new MultiFileSource(dataPath));
    return trainingDataView;
}
```

Simple to build classifier

```
var pipeline = new TextTransform(env, "Text", "Features")
                .Append(new LinearClassificationTrainer(env, new
LinearClassificationTrainer.Arguments(),
    "Features",
    "Label"));
var model = pipeline.Fit(trainingData);
```

Evaluation of Models

```
PredictionModel quality metrics evaluation
------------------------------------------
Accuracy: 94.44%
Auc: 98.77%
F1Score: 94.74%
```

```
IDataView testData = GetData(env, _testDataPath);

var predictions = model.Transform(testData);

var binClassificationCtx = new BinaryClassificationContext(env);
var metrics = binClassificationCtx.Evaluate(predictions, "Label");
```

Demo

Loading and Transforming data in a pipeline

DATA

Supported Types

- Text (CSV/TSV)
- Parquet
- Binary
- IEnumerable<T>
- File sets

The problem with flat files

- With no database or storage engine Data is written arbitrarily to disc

- Format errors
  - Caused by bug
  - Caused by error

- Inefficiencies
  - Compression an afterthought
  - GZIP splittable problem

More problems with flat files

- Access errors
  - Mutability
  - Variability between files in a fileset
- Naivety
  - Just because brute force scanning is possible doesn't mean it's optimal
  - Predicate Push-Down
  - Partitioning

Parquet

- Apache Parque is a file format, primarily driven from Hadoop Ecosystem, particularly loved by Spark
- Columnar format

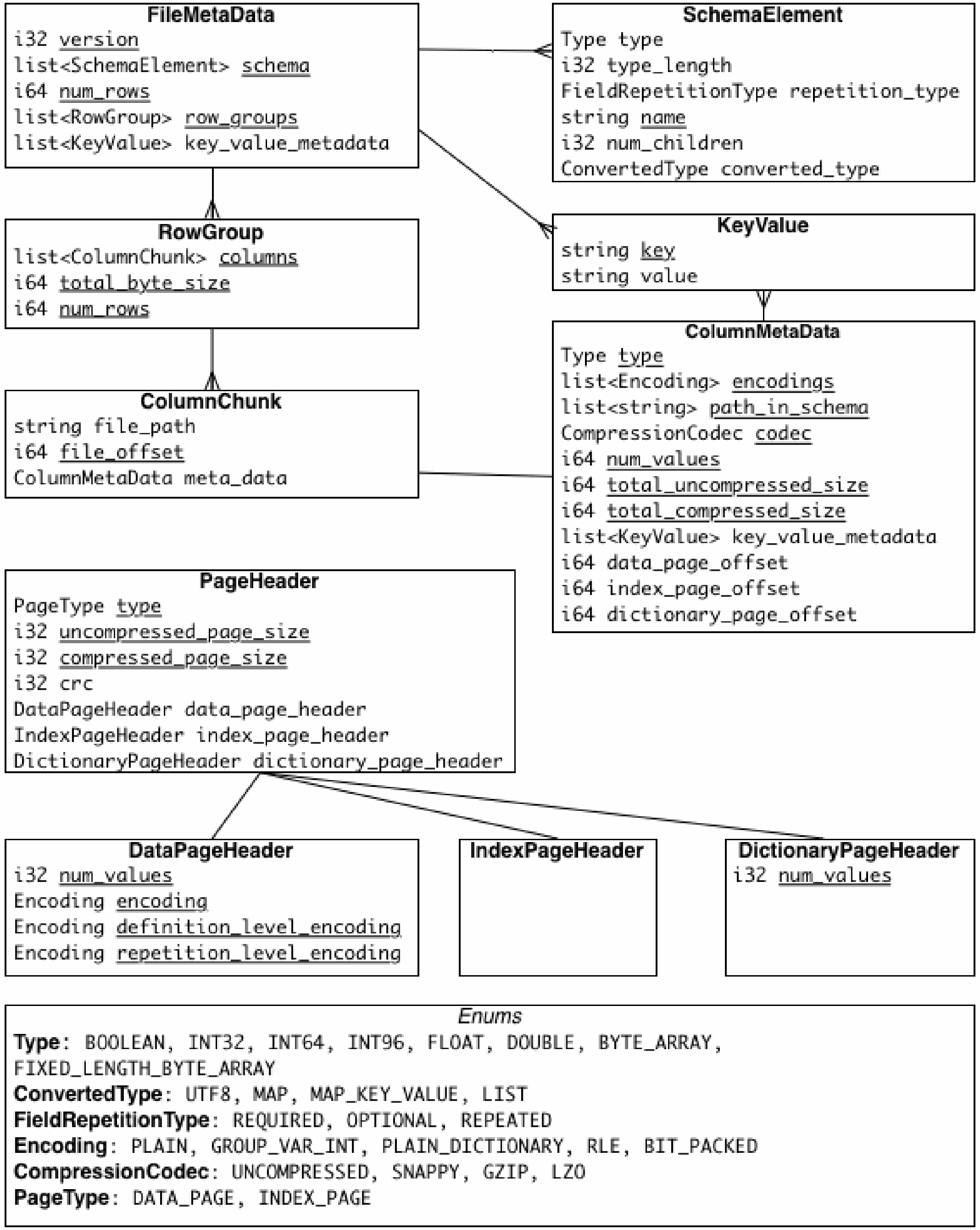
- Block
  - File
    - Row Group
      - Column Chunk
        - » Page

Designed for Data(!)

- Schemas consistent through file
- Held at the end so can be quickly seeked

- By Convention: WRITE ONCE

- Parallelism:
  - Partition workload by File or Row Group
  - Read data by column chunk
  - Compress per Page

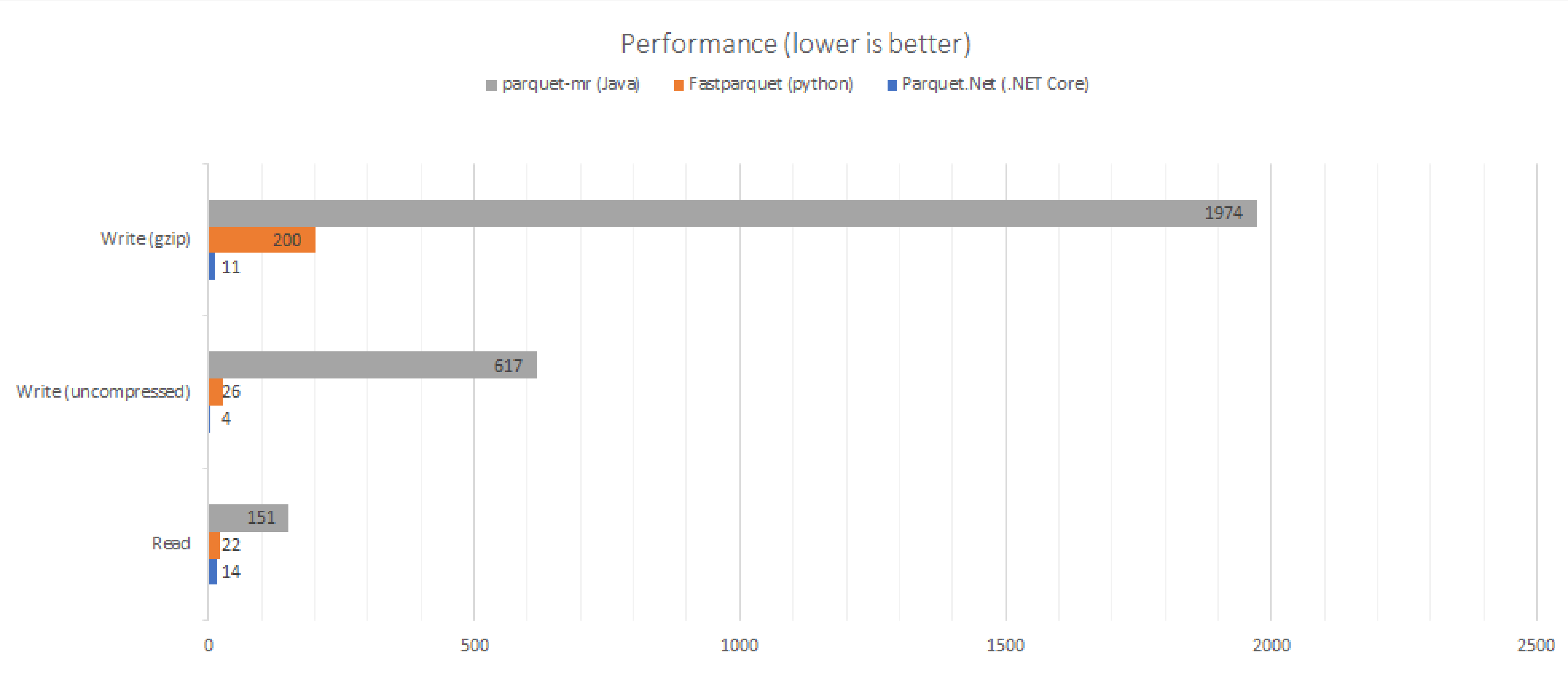

FileMetaData
i32 version
list<SchemaElement> schema
i64 num_rows
list<RowGroup> row_groups
list<KeyValue> key_value_metadata
SchemaElement
Type type
i32 type_length
FieldRepetitionType repetition_type
string name
i32 num_children
ConvertedType converted_type
RowGroup
list<ColumnChunk> columns
i64 total_byte_size
i64 num_rows
KeyValue
string key
string value
ColumnChunk
string file_path
i64 file_offset
ColumnMetaData meta_data
ColumnMetaData
Type type
list<Encoding> encodings
list<string> path_in_schema
CompressionCodec codec
i64 num_values
i64 total_uncompressed_size
i64 total_compressed_size
list<KeyValue> key_value_metadata
i64 data_page_offset
i64 index_page_offset
i64 dictionary_page_offset
PageHeader
PageType type
i32 uncompressed_page_size
i32 compressed_page_size
i32 crc
DataPageHeader data_page_header
IndexPageHeader index_page_header
DictionaryPageHeader dictionary_page_header
DataPageHeader
i32 num_values
Encoding encoding
Encoding definition_level_encoding
Encoding repetition_level_encoding
IndexPageHeader
DictionaryPageHeader
i32 num_values
Enums
Type: BOOLEAN, INT32, INT64, INT96, FLOAT, DOUBLE, BYTE_ARRAY, FIXED_LENGTH_BYTE_ARRAY
ConvertedType: UTF8, MAP, MAP_KEY_VALUE, LIST
FieldRepetitionType: REQUIRED, OPTIONAL, REPEATED
Encoding: PLAIN, GROUP_VAR_INT, PLAIN_DICTIONARY, RLE, BIT_PACKED
CompressionCodec: UNCOMPRESSED, SNAPPY, GZIP, LZO
PageType: DATA_PAGE, INDEX_PAGE

Parquet.NET

- Until recently no approach available to .NET
  - Leading to System.IO to write arbitrary data and then requiring data engineering to sort the data out

- Libraries for cpp can be used
  - Implementation by G-Research called ParquetSharp uses Pinvoke

- A full .NET Core implementation is Parquet.NET
  - https://github.com/elastacloud/parquet-dotnet


Performance (lower is better)
parquet-mr (Java)
Fastparquet (python)
Parquet.Net (.NET Core)
Write (gzip)
Write (uncompressed)
Read
1974
200
11
617
26
4
151
22
14
0
500
1000
1500
2000
2500

Parq!

- End to end tooling for .NET devs on a platform they're familiar with.

- Uses dotnet global tooling

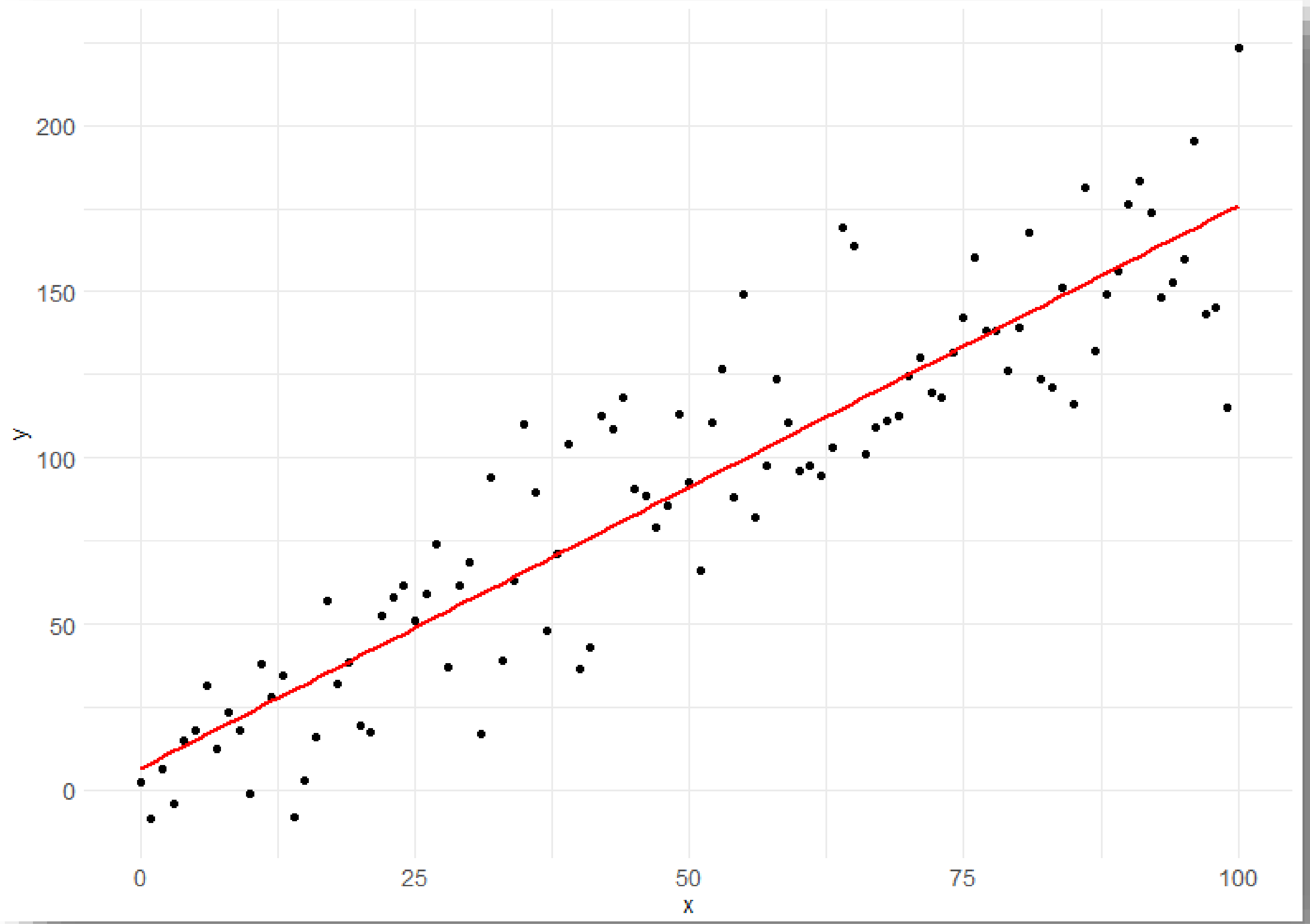
DEMO OF PARQUET.NET

REGRESSION ANALYSIS

What is regression?

- Supervised Machine Learning
- Features go to a Label
- Label is a “real value” not a categorical like in classification

- Regression algorithms generate weights for features


200
150
100
50
0
y
0
25
50
75
100
x

Taxi Fare Data

- NYC data
- Taxi Fares, Time, Distance, Payment Method etc
- Use these as features and predict the most likely Fare ahead of time.

```
var pipeline = new LearningPipeline
{
    new TextLoader(_dataPath).CreateFrom<TaxiTrip>(useHeader: true, separator: ','),
    new ColumnCopier(("FareAmount", "Label")),
    new CategoricalOneHotVectorizer(
        "VendorId",
        "RateCode",
        "PaymentType"),
    new ColumnConcatenator(
        "Features",
        "VendorId",
        "RateCode",
        "PassengerCount",
        "TripDistance",
        "PaymentType"),
    new FastTreeRegressor()
};
```

```
PredictionModel<TaxiTrip, TaxiTripFarePrediction> model = pipeline.Train<TaxiTrip, TaxiTripFarePrediction>();
```

Demo

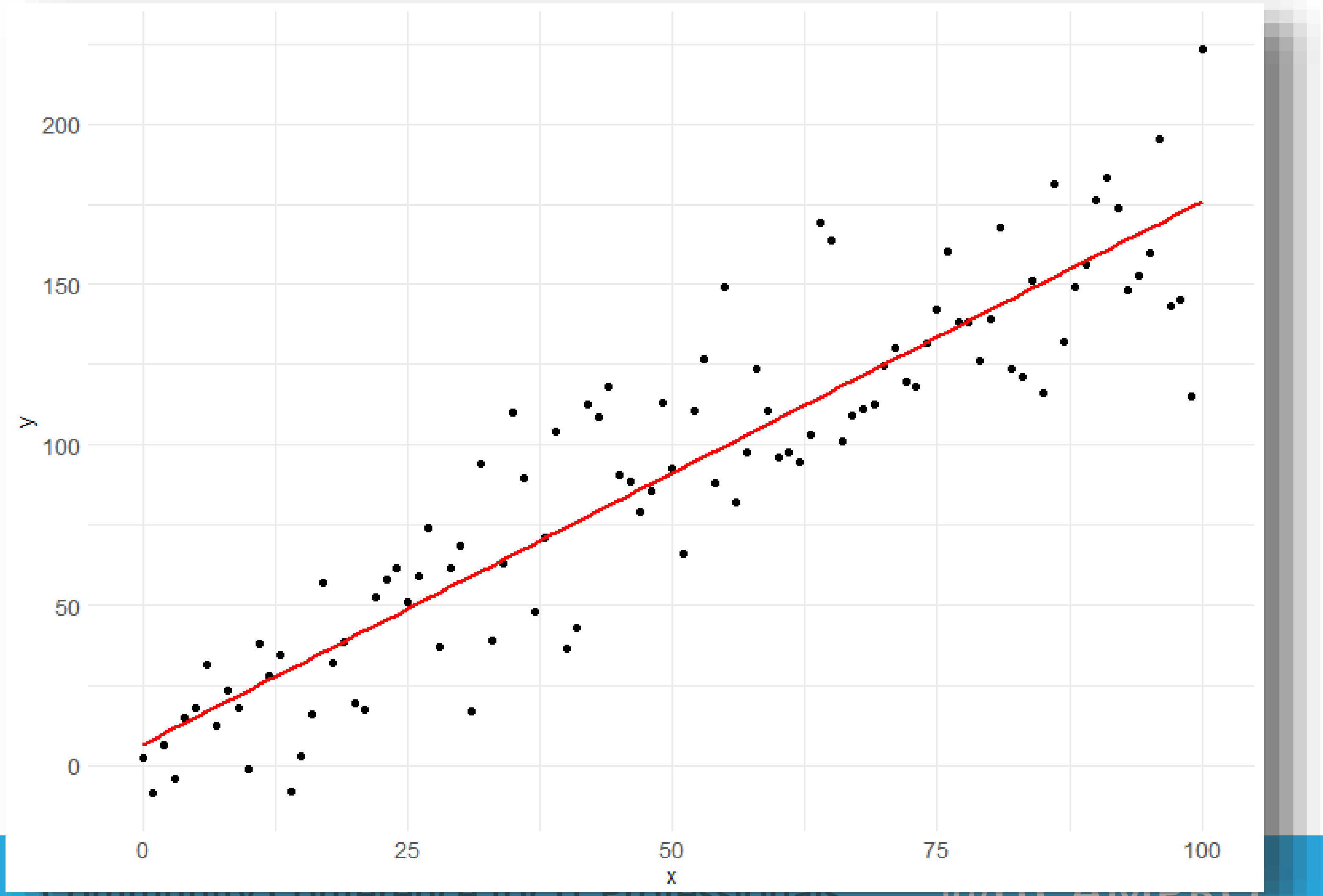
Evaluation with RMS and r^2

```
Rms = 3.30299146626885
RSquared = 0.885729301000846
Predicted fare: 31.14972, actual fare: 29.5
```

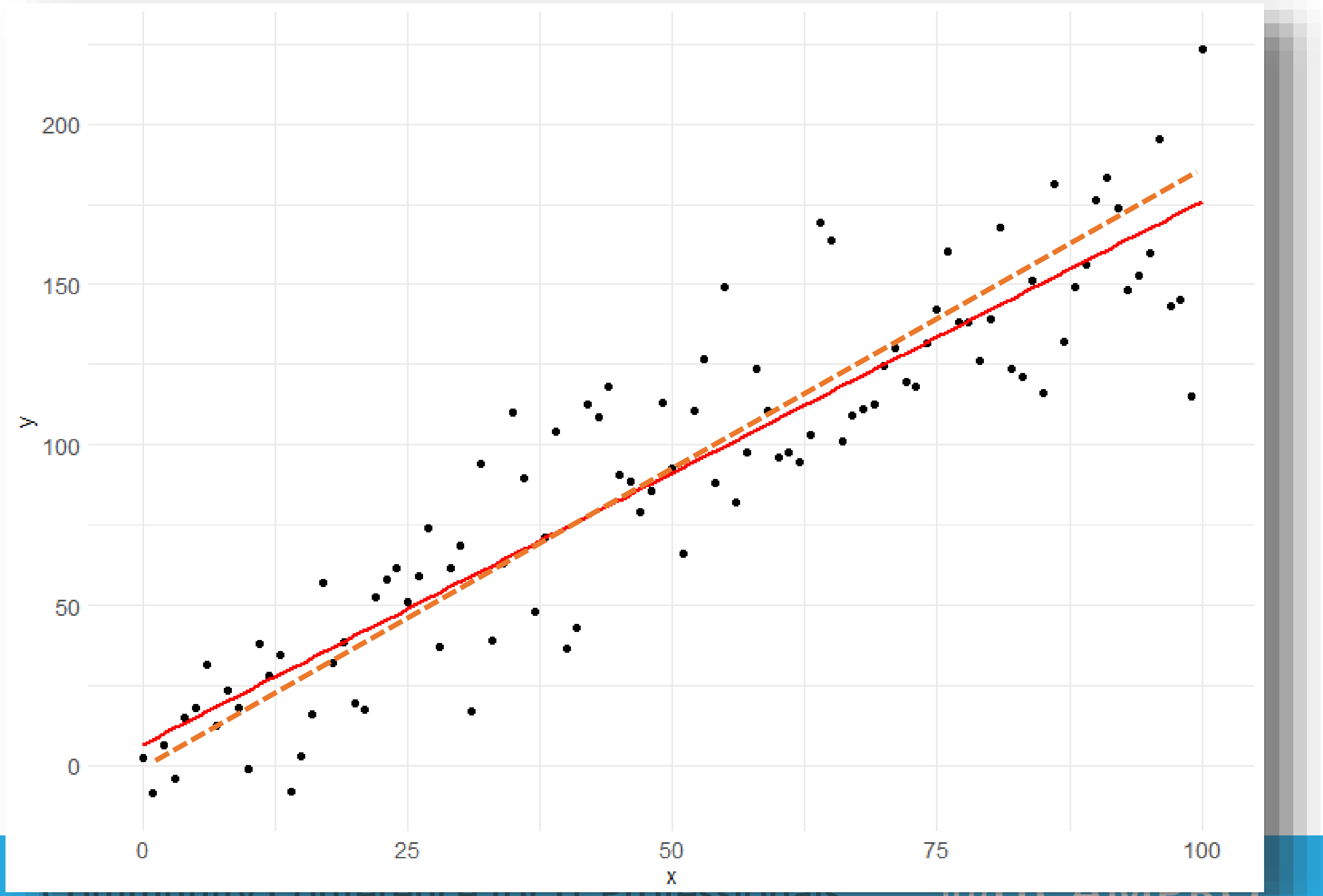
r^2 is the proportion of the variance in the dependent variable that is predictable from the independent variables

COEFFICIENT OF DETERMINATION

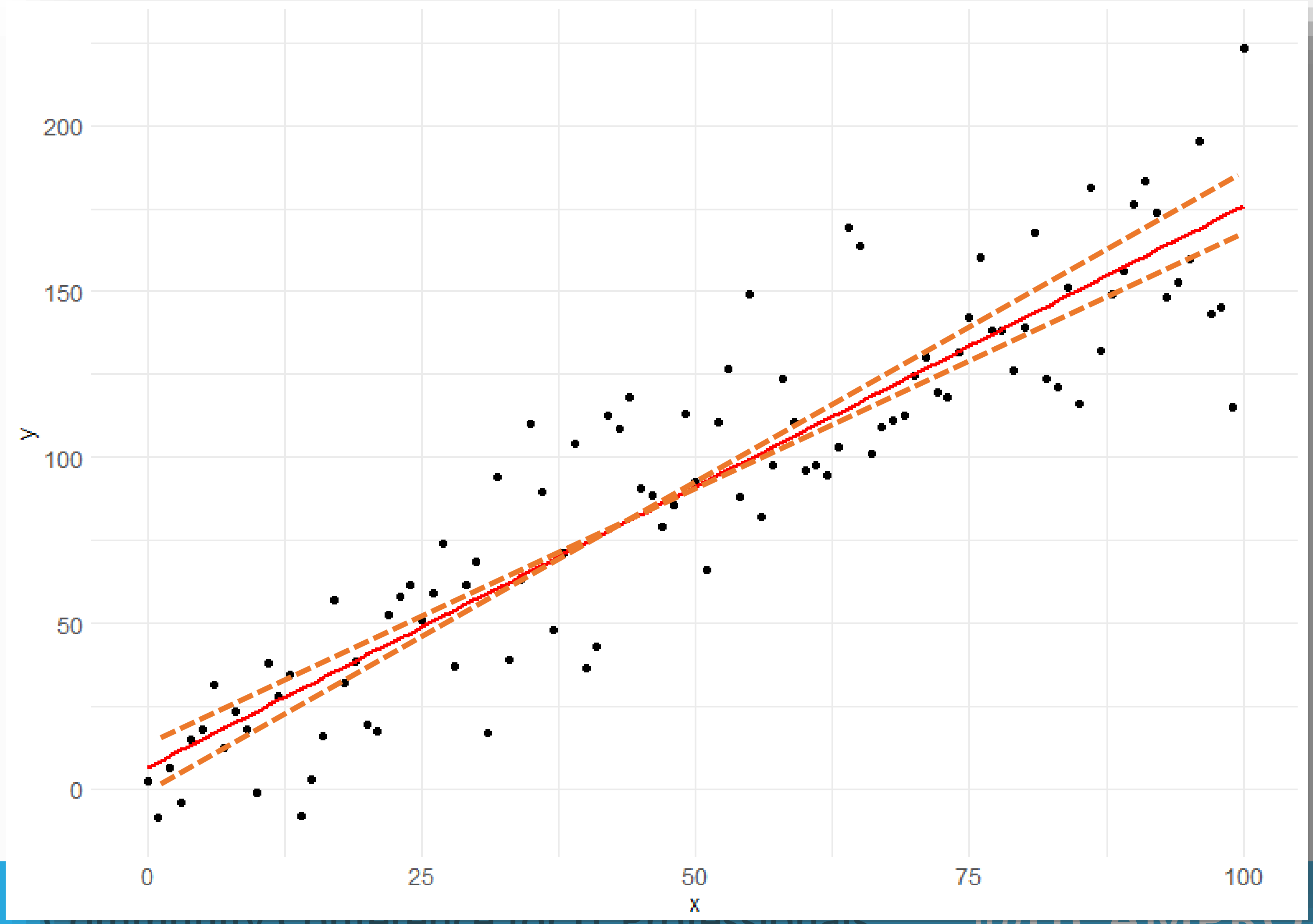
How good a fit is my line?



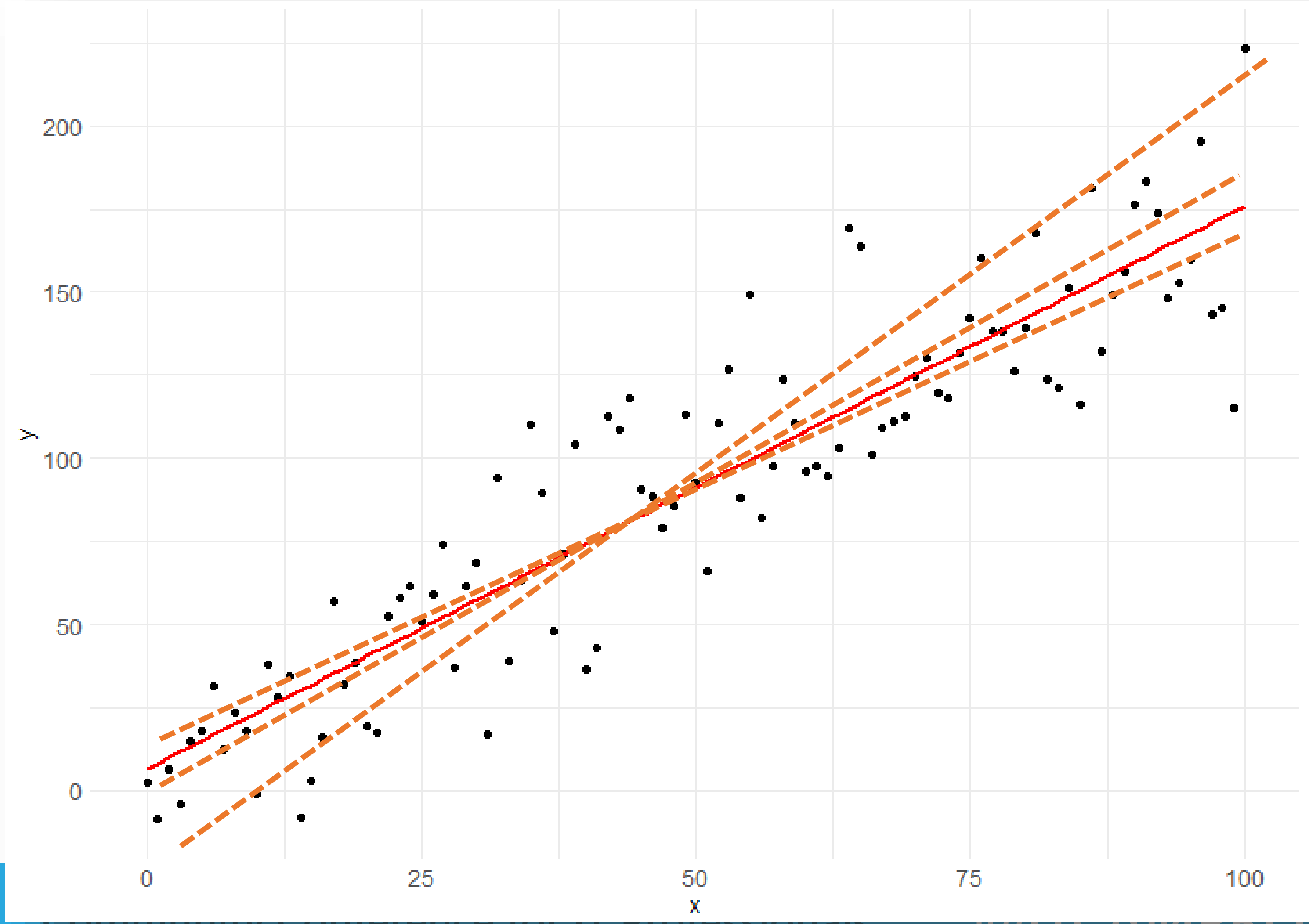
Compared to....



Compared to....



Compared to....



What does it tell us?

- In a multiple-regression model to determine solar energy production:
  - The *energy production* is the dependent variable (Y)
  - The *cloud cover level* is an independent variable (X1)
  - The *season of year* is an independent variable (X2)

  - Y = X1*weightX1 + X2*weightX2

- It's a coefficient (ratio) of how good my predictions are versus the amount of variability in my dependent variable.

How does it do that?

- It measures the relationship between two variables:
  - Y-hat : ŷ
    - The estimator is based on regression variables, Intercept, X Variable 1 to X Variable n
    - The distance from this estimator (prediction) and the real y value (y- ŷ)
    - Squared
  - Y-bar : ȳ
    - The average value of all Ys
    - The distance of the real y value from the mean of all y values (y-ȳ), which is how much the data varies from average
    - Squared
- These two squares are summed, and calculated:
  - 1-(((y-ŷ)^2)/((y-ȳ)^2))
  - 1-(estimation error/actual distance from average)

Y is our actual value: *Energy generation*

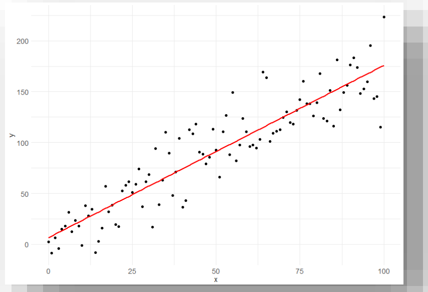
X1 is our first input value we want to predict from, X2 our second: *Irradiance Percentage* and *Days left until service*

Y	X1	X2
973	0	40
1119	0	40
875	25	25
625	25	25
910	30	30
971	30	30
931	35	35
1177	35	35
882	40	25
982	40	25
1628	45	45
1577	45	45
1044	50	0
914	50	0
1329	55	25
1330	55	25
1405	60	30
1436	60	30
1521	65	35
1741	65	35
1866	70	40
1717	70	40

sum	26953	950	660
mean	1225	43	30
sd	346	20	12

	Coefficients
Intercept	156.43
X Variable 1	13.0807
X Variable 2	16.7953

The data science team works hard to understand the data and model it, which means produce weights for how much each input variable affects the actual value



If you were to plot out these values

Intercept + X1 * X Variable 1 + X2 * X Variable 2
(weightX1) (weightX2)

You would get a straight line like the red one above

To calculate a prediction (called y-hat) we add the intercept to the variable values multiplied by their modifiers (coefficient)

=$N$20+$N$21*[@X1]+$N$22*[@X2]

B	C	D	E	F	G
	X1	X2	ŷ	y-ŷ	y-ȳ
973	0	40	*[@X2]	145	-25

	Coefficients
Intercept	156.43
X Variable 1	13.0807
X Variable 2	16.7953

ŷ
828.242
828.242
903.329
903.329
1052.71
1052.71
1202.09
1202.09
1099.54
1099.54
1500.85
1500.85
810.464
810.464
1295.75
1295.75
1445.13
1445.13
1594.51
1594.51
1743.89
1743.89

IF we take away the prediction (estimator) from the actual value we have the residual, or the size of the error. If this is too high, the error is positive, if the number is too low, the error is negative.

You might sometimes hear data scientists talking about residuals; these are the residuals.

It is simply the actual value minus the prediction.

=[@Y]-[@ŷ]

	E	F
	ŷ	y-ŷ
40	828.242	145
40	828.242	291
25	903.329	-28
25	903.329	-278
30	1052.71	-143
30	1052.71	-82
35	1202.09	-271
35	1202.09	-25
25	1099.54	-218
25	1099.54	-118
45	1500.85	127
45	1500.85	76
0	810.464	234
0	810.464	104
25	1295.75	33
25	1295.75	34
30	1445.13	-40
30	1445.13	-9
35	1594.51	-74
35	1594.51	146
40	1743.89	122
40	1743.89	-27
660		0.00

=[@Y]-$B$25

mean	1225

y-ȳ
-252
-106
-350
-600
-315
-254
-294
-48
-343
-243
403
352
-181
-311
104
105
180
211
296
516
641
492
0.00

We can also measure the difference between the observed actual value and the average (mean) of the whole set of actuals.

This gives us a distance measure of variance, how far the actual varies from the average value of the actuals. This is a measure of the variance of the data.

If the number is bigger than average, the number is positive, if it is smaller than average, the number is negative.

=[@[y-ŷ]]^2

=[@[y-ȳ]]^2

Both the size of the measures we just created have a sum of 0, because some values are bigger and some smaller than the actuals. This means when they get added up they cancel out to 0.

This is correct, but we are trying to compare the sizes of the errors, not whether they are above or below actual, so we need to lose the sign and make everything positive.

The way we'll do that is by times-ing the number by itself (squaring the number), as -1*-1 = 1, -2 * -2 = 4 etc

Adding all these up across the set gives us the sums of squares

H	I
(y-ŷ)^2	(y-ȳ)^2
20955.00662	63572.74587
84540.47177	11264.92769
802.5554941	122595.4731
77467.2607	360163.655
20365.91736	99310.92769
6676.394277	64585.29132
73489.24633	86516.20041
629.4579591	2317.109504
47323.48833	117742.564
13815.56339	59115.29132
16167.48078	162299.1095
5799.037076	123808.0186
54538.83802	32810.38223
10719.60342	96805.83678
1105.573201	10787.65496
1173.073523	10996.38223
1610.387545	32350.92769
83.35022164	44463.47314
5403.636215	87535.29132
21459.48727	266115.2913
14911.04147	410706.2004
723.0303999	241929.8368
479759.90	2507792.59

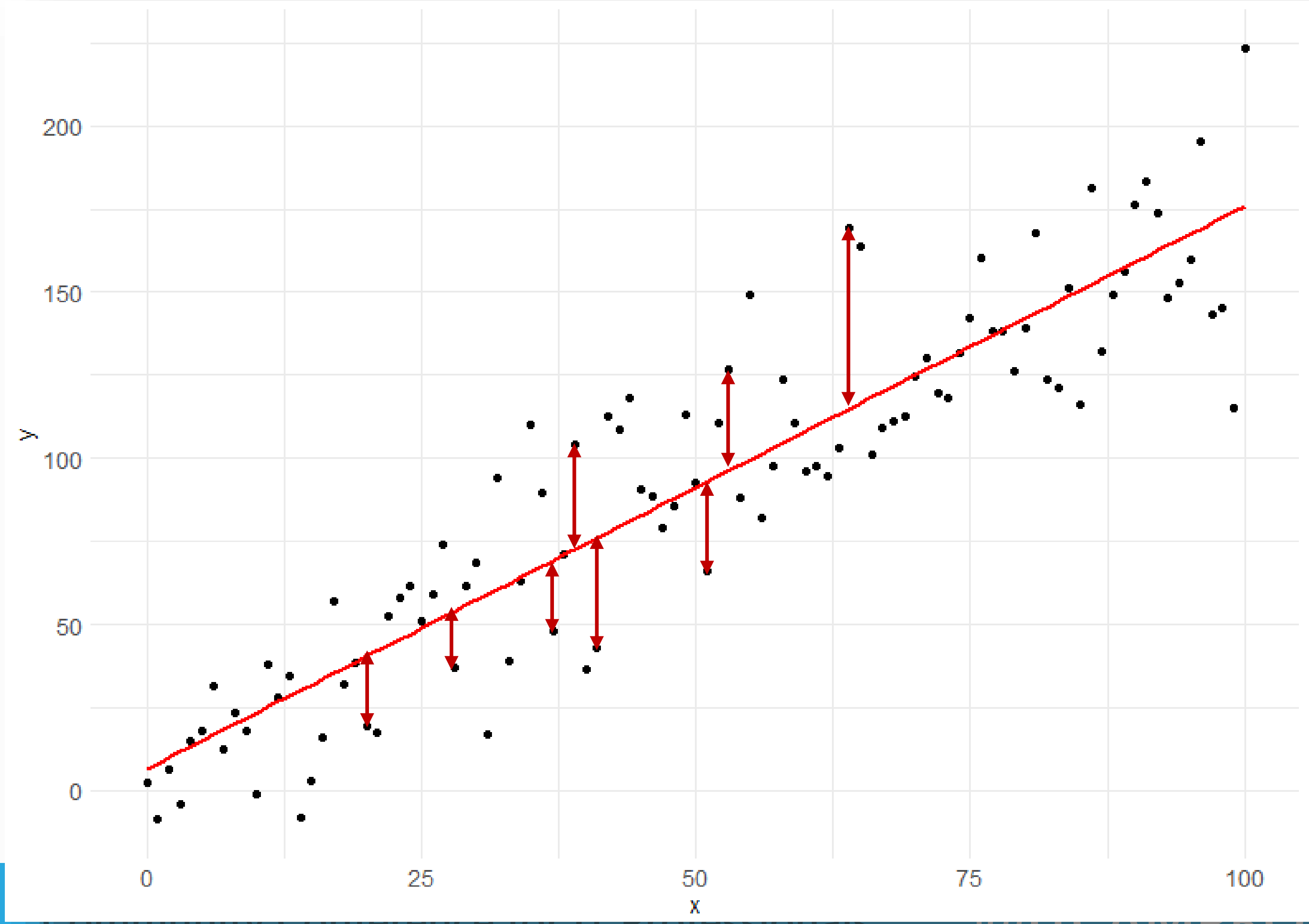
We then divide the sum of squares of our estimator error by the sum of squares of our distance from average. The estimator errors are always lower than the variance of the data, and we take the result from 1, to give us a value like 0.80, which we shouldn't describe as 80% accurate, but you can think of it along these lines; 80% of variance is explained by the model.

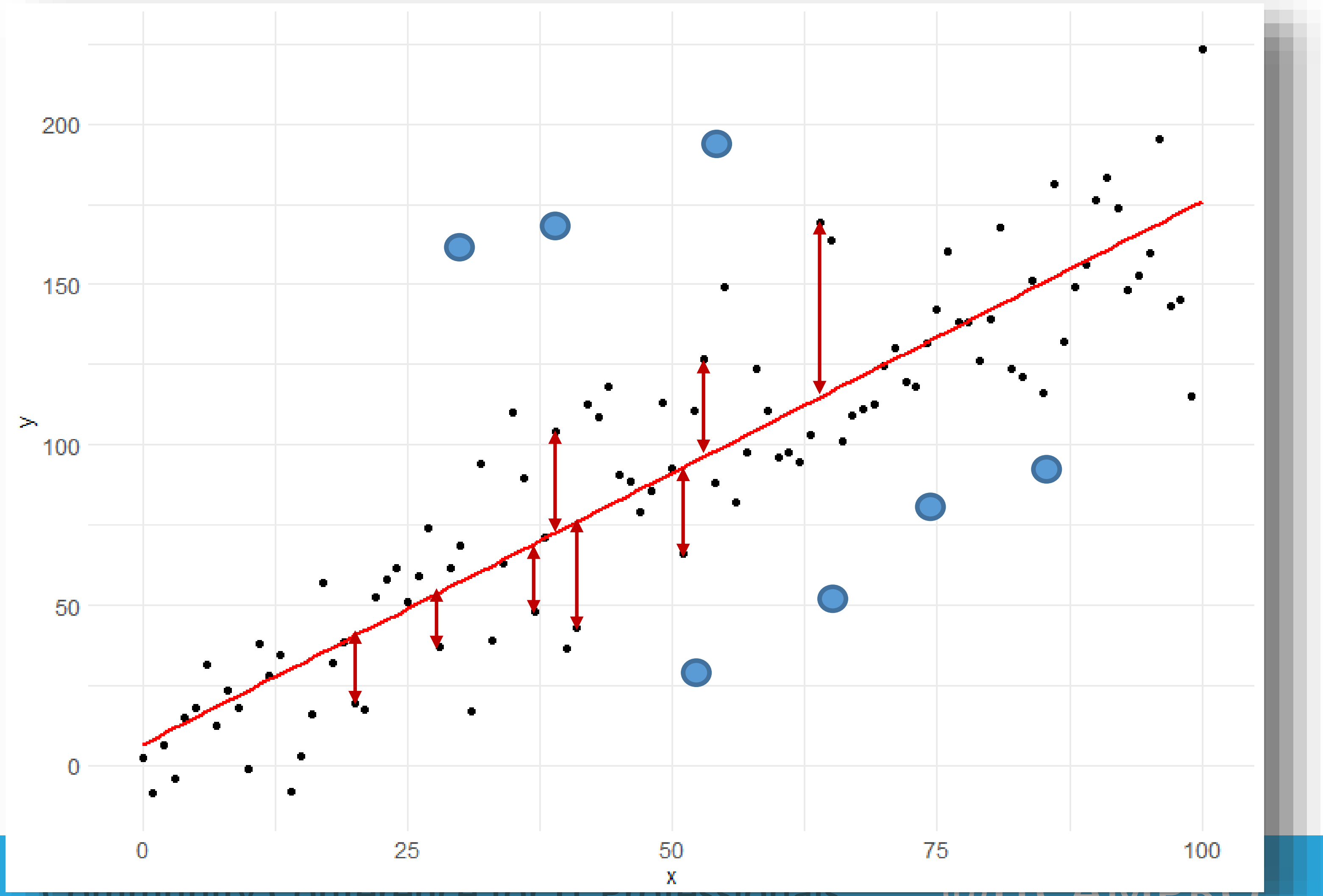
479759.90	2507792.59		
		r2	0.80869

=1-H24/I24

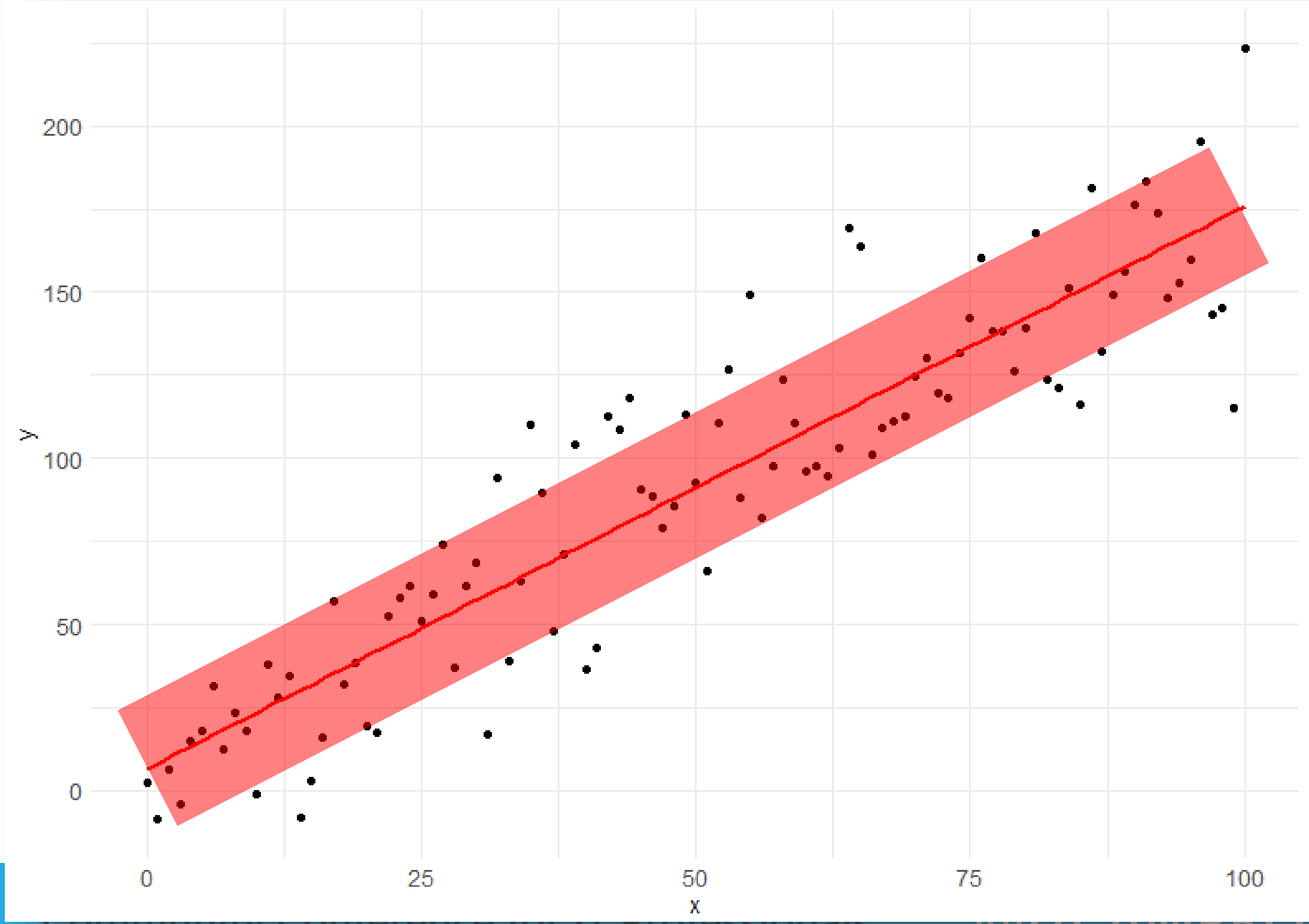
RMSE is the average of how far out we are on an estimation by estimation basis

ROOT MEAN SQUARED ERROR


200
150
100
50
0
y
0
25
50
75
100
x



200
150
100
50
0
y
0
25
50
75
100
x


200
150
100
50
0
y
0
25
50
75
100
x

What does it tell us?

- By comparing the average distance between the estimator ŷ and the observed actual, we can get a measure of how close in real world terms we are on average to the actual.
- Unlike r^2 which gives an abstract view on variance, RMSE gives the bounds to the accuracy of our prediction.
- On average, the real value is +/- RMSE from the estimator.

How do we calculate it?

- It's very easy and we've already done the hard work.

- The MSE part means average of the squared distance of the estimator from the actual
  - =AVERAGE(data[(y-ŷ)^2])

- Since the values were squared, this number is still big; square rooting this gives us a real world value.
  - =SQRT(AVERAGE(data[(y-ŷ)^2]))

Take the average of the squared estimator distance from actual.

Squaring it earlier was useful, as the non-squared value averages out to zero!

I	J	K	L	M
(y-ŷ)^2	(y-ȳ)^2			
20955.00662	63572.74587			
84540.47177	11264.92769			
802.5554941	122595.4731			
77467.2607	360163.655			
20365.91736	99310.92769			
6676.394277	64585.29132			
73489.24633	86516.20041			
629.4579591	2317.109504			
47323.48833	117742.564			
13815.56339	59115.29132			
16167.48078	162299.1095			
5799.037076	123808.0186			
54538.83802	32810.38223			
10719.60342	96805.83678			
1105.573201	10787.65496			
1173.073523	10996.38223			
1610.387545	32350.92769			
83.35022164	44463.47314			
5403.636215	87535.29132			
21459.48727	266115.2913			
14911.04147	410706.2004			
723.0303999	241929.8368			
479759.90	2507792.59			
			MSE	21807.3

Since the MSE is still in the order of magnitude of the Squares, square root it to give us a real world value and this is Root Mean Square Error (RMSE).

```
=SQRT(M25)
```

MSE	21807.3	rmse	147.673

In this example, the estimate is on average 147.673 away from the actual value.

THIS MODEL CAN THEREFORE BE DESCRIBED AS:

BEING ABLE TO DESCRIBE 80% OF VARIANCE

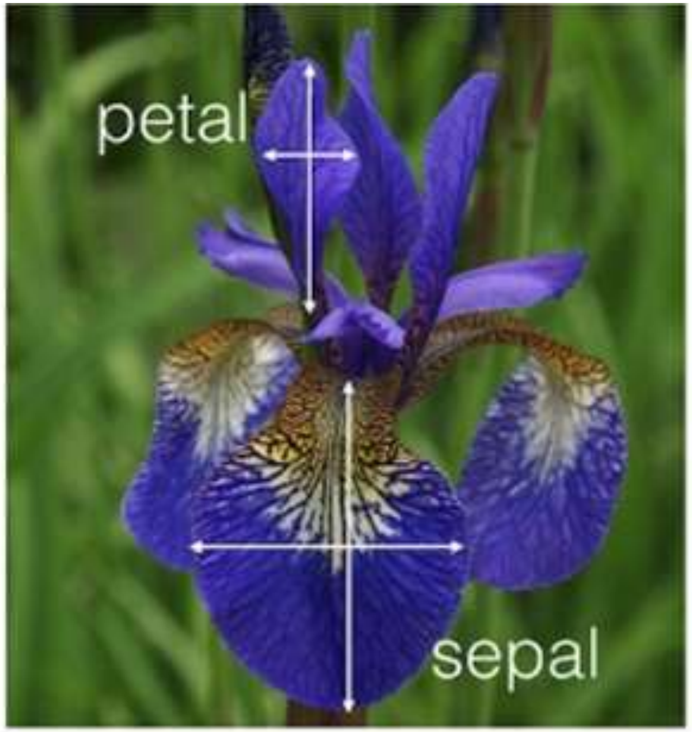
Example of the Iris petal detector

CLUSTERING

What is Clustering

- Used to generate groups or clusters of similarities
- Identify relationships that are not evident to humans
- Examples may be:
  - Who is a VIP customer?
  - Who is cheating in a game?
  - How many people are likely to drop out of class?
  - Which components in manufacturing are likely to fail?

Example: Iris flower



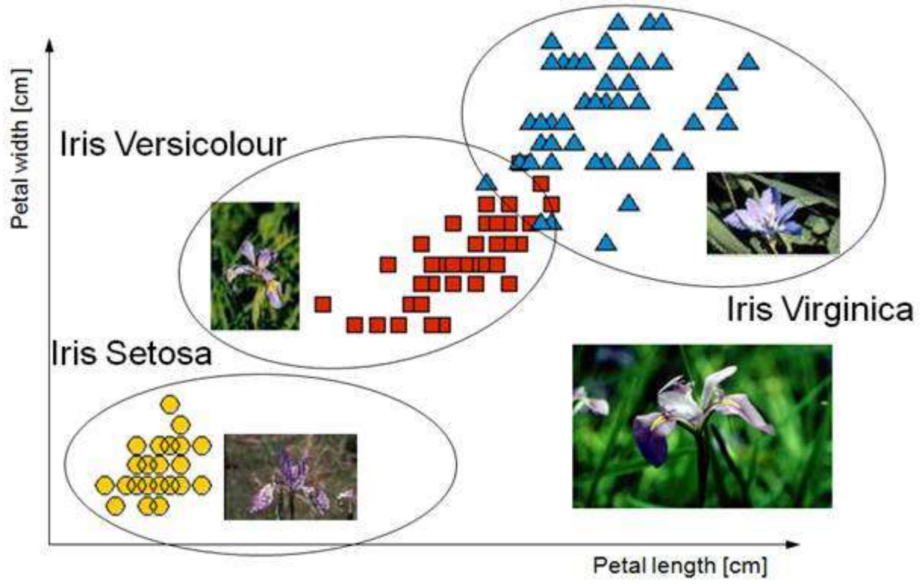
Features				Labels
Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	Iris setosa
4.9	3.0	1.4	0.2	Iris setosa
7.0	3.2	4.7	1.4	Iris versicolor
6.4	3.2	4.5	1.5	Iris versicolor
6.3	3.3	6.0	2.5	Iris virginica
5.8	3.3	6.0	2.5	Iris virginica

Iris Versicolor Iris Setosa Iris Virginica

Supervised Learning

Definition
- You give the input data (X) and an output variable (Y) (labels), and you use an algorithm to learn the mapping function from the input to the output. Y = f(X)
Techniques
- Classification: you want to classify a new input value. - Regression

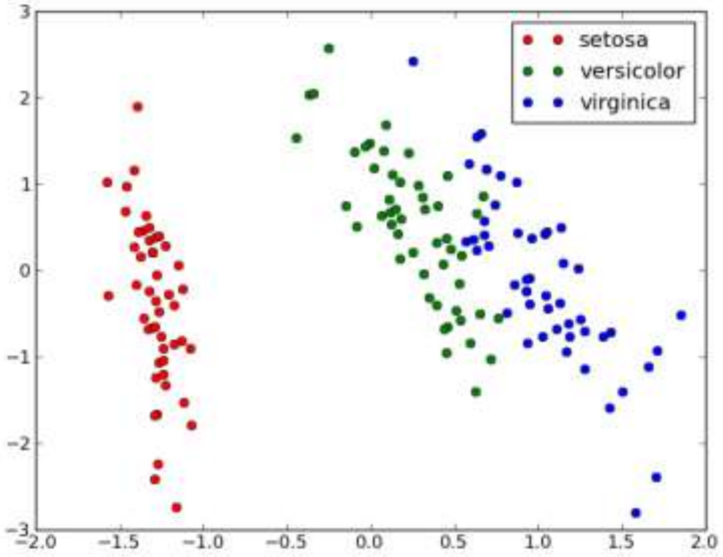
Features				Labels
Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	Iris setosa
4.9	3.0	1.4	0.2	Iris setosa
7.0	3.2	4.7	1.4	Iris versicolor
6.4	3.2	4.5	1.5	Iris versicolor
6.3	3.3	6.0	2.5	Iris virginica
5.8	3.3	6.0	2.5	Iris virginica



Unsupervised Learning

Definition
- You give the input data (X) and no corresponding output variables (labels).
Techniques
- Clustering: you want to discover the inherent groupings in the data. - Association: you want to discover rules that describe large portions of your data.

Features				~~Labels~~
Sepal length	Sepal width	Petal length	Petal width	~~Species~~
5.1	3.5	1.4	0.2	~~Iris setosa~~
4.9	3.0	1.4	0.2	~~Iris setosa~~
7.0	3.2	4.7	1.4	~~Iris versicolor~~
6.4	3.2	4.5	1.5	~~Iris versicolor~~
6.3	3.3	6.0	2.5	~~Iris virginica~~
5.8	3.3	6.0	2.5	~~Iris virginica~~



As easy as omitting the input label

```
pipeline.Add(new ColumnConcatenator(
            "Features",
            "SepalLength",
            "SepalWidth",
            "PetalLength",
            "PetalWidth"));
```

Choose a clusterer

```
pipeline.Add(new KMeansPlusPlusClusterer() { K = 3 });
```

Hyperparameters

Demo

Loading trained models from other frameworks

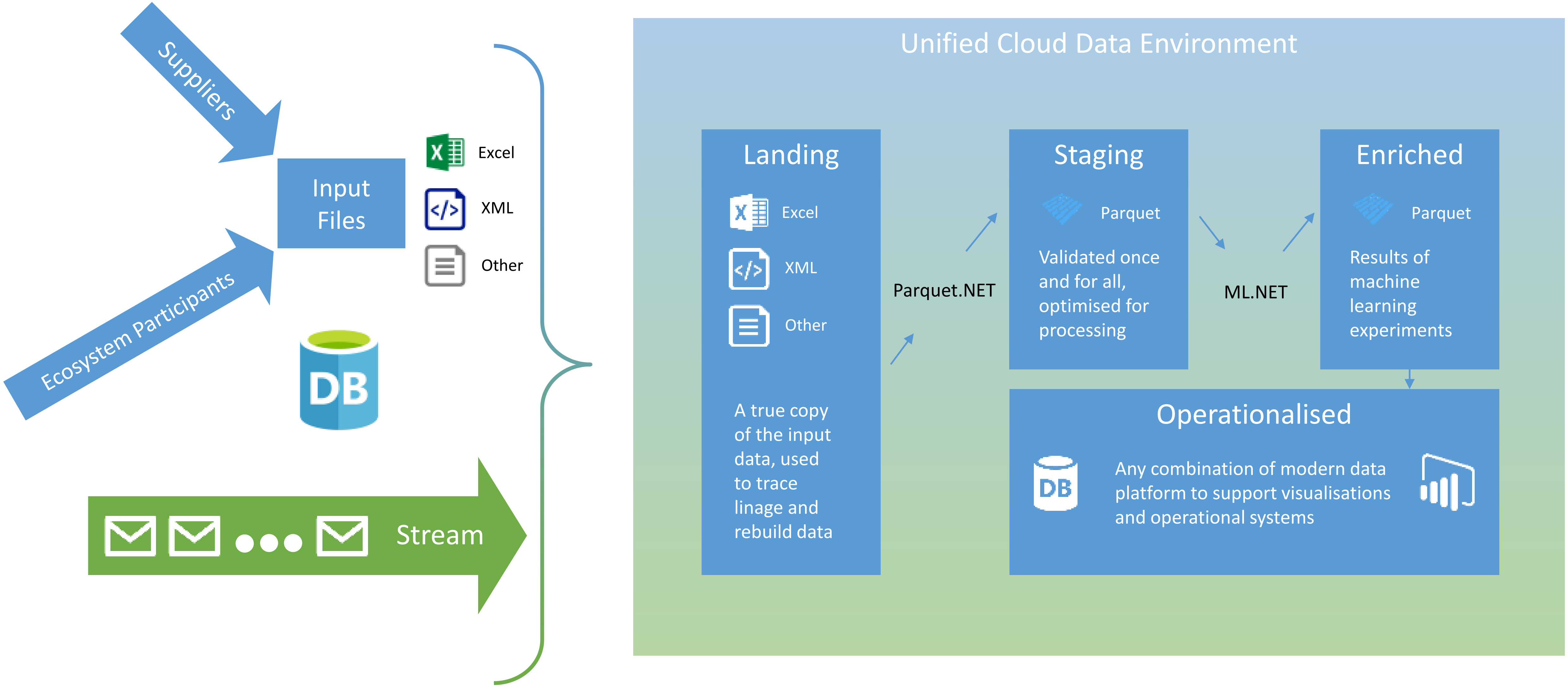
INTEROPERABILITY

TensorFlow since 0.5.0

- Additional work for:
  - CNTK
  - Torch
- This means ML.NET can be the OSS, XPLAT host for many data science frameworks, anywhere that NETCORE runs.
- On devices with Xamarin, on edge devices, on web servers.

Demo

SUMMARY



Suppliers
Ecosystem Participants
Input Files
Excel
XML
Other
DB
Stream
Unified Cloud Data Environment
Landing
Excel
XML
Other
A true copy of the input data, used to trace linage and rebuild data
Parquet.NET
Staging
Parquet
Validated once and for all, optimised for processing
ML.NET
Enriched
Parquet
Results of machine learning experiments
Operationalised
DB
Any combination of modern data platform to support visualisations and operational systems

EPILOGUE: THE BUSINESS OF ML

The questions to answer

- Opportunity discovery
  - What can we do?
- Adoption
  - How do I get my company to use this?
- Sustainability
  - How do we control costs and/or build a viable price point?
- Measurement
  - How do we know it's generating value?

Build

Use

Profit

The end goals

- Providing better customer service and experience
  - Better products
  - Self-healing and self-improving
- Improving operational processes and resource consumption
  - Faster Time to Market
  - Lighter Bill of Materials
  - Stronger Supply Chain
- New business models
  - Insights from data
  - Different pricing models

Dive deeper into strategic modelling with these mental models

MENTAL MODELS

Mental Models

In order to envision the transformation possibilities, adopt the following mental models.

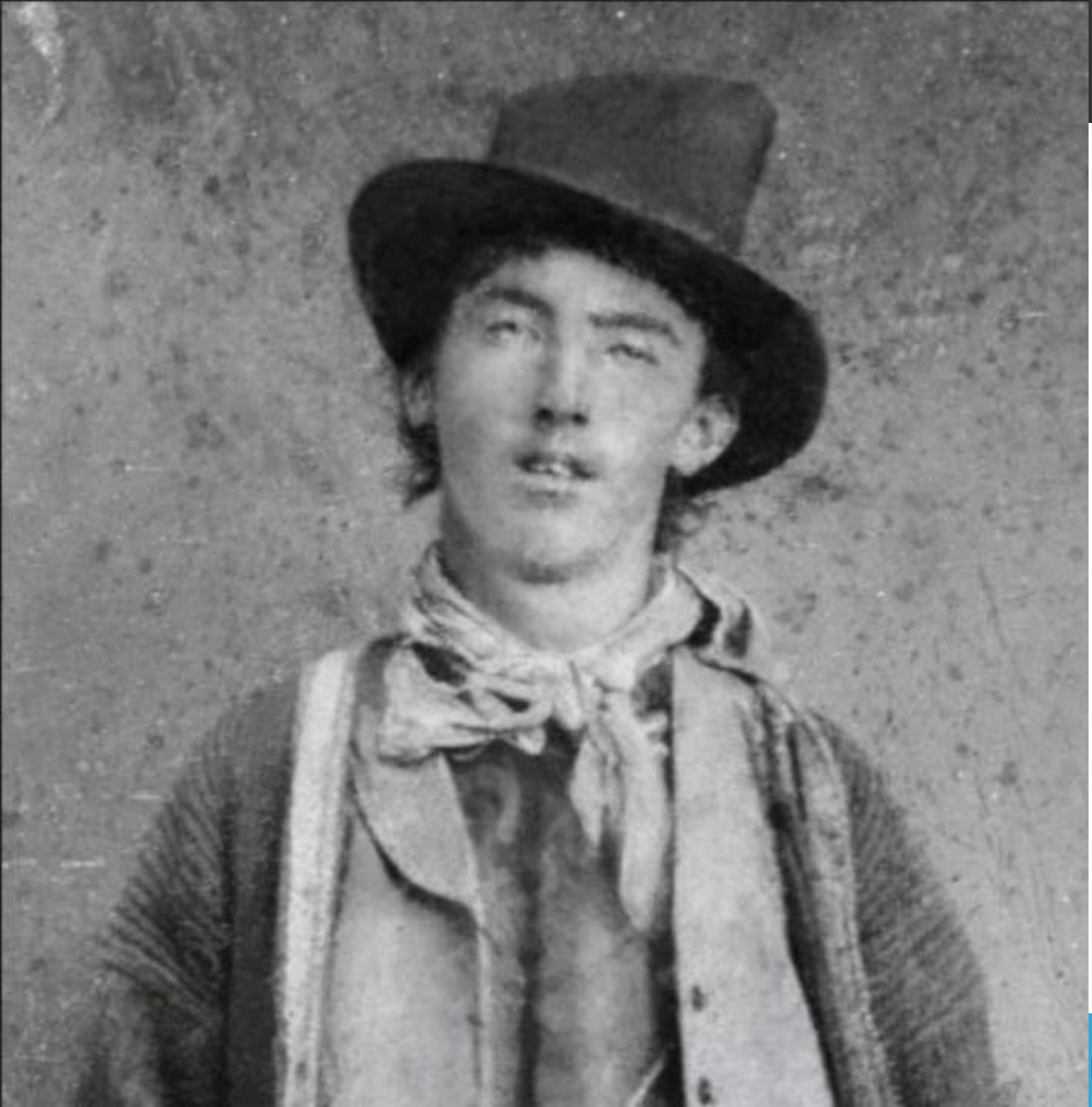
AI is an extremely focussed entry level clerical assistant

AI is exceptionally reactive

AI is auditable, repeatable, improvable

AI is resistant of external bias

One Million Clerical Workers
What is achievable with a million trained office workers?



Billy the Kid Reaction Times

What if all workers can react and form a course of action in 1 millisecond?

Great companies have high cultures of accountability, it comes with this culture of criticism […], and I think our culture is strong on that.

Steve Ballmer

Cement the views of the Founders

What if the decision making process of the hyper-

REGARDING MAINTENANCE CONTRACTS

JIT Service Regimes

- Migrate from manufacturer led maintenance cycles to just-in-time
- Use data to establish mean time to failure
- Conditional maintenance

Alarm

- Predictions through AI become an integral part of business reporting

~ 1 minute telemetry

through dedicated cloud gateway

AzureML

Reporting

- Machine Learning used in real-time with “Anomaly Detection” which enables history to be assessed and false positive spikes in behaviour to be discarded
- Model retraining can occur to help understand acceptable lowering of efficiency as equipment degrades over time within acceptable parameters

Supplier efficiency

- Use data to judge the reliability of assets
- Track fault recurrence and resolution speeds
- Contract with SLAs around reducing outages

WAREHOUSE AND INVENTORY MANAGEMENT

JIT Warehousing

- **Regarding Parts**
- JIT Maintenance leads to JIT Warehousing

- **Regarding Fuel**
- Integrated Data streams including ERP and CRM
- Manage the supply chain and keep the right stock level

G4S Data Science Programme

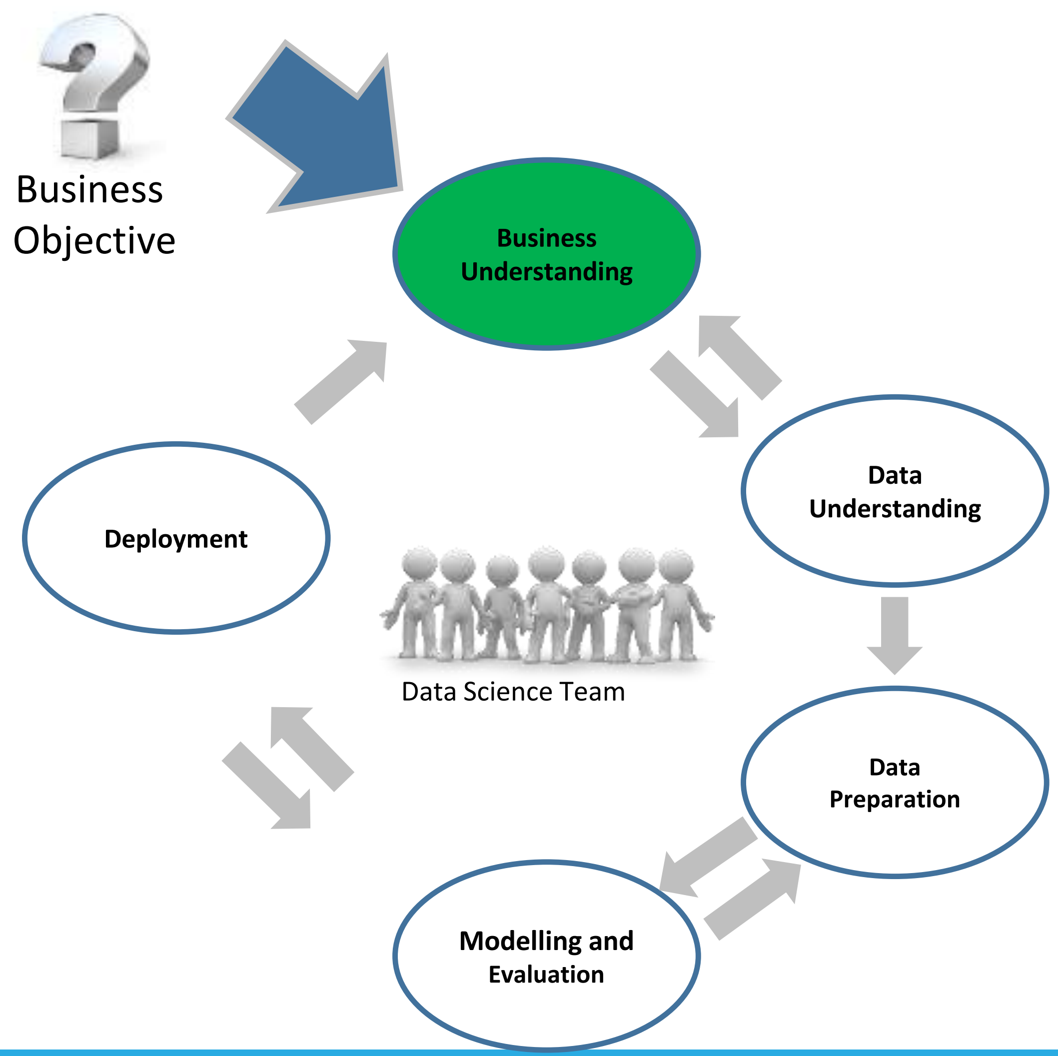
- G4S plc is a British multinational security services company and operates an integrated security business in more than 90 countries across the globe.

- They aim to differentiate G4S by providing industry leading security solutions that are innovative, reliable and efficient.

- activity in

EPILOGUE II: RUNNING A DSCI TEAM

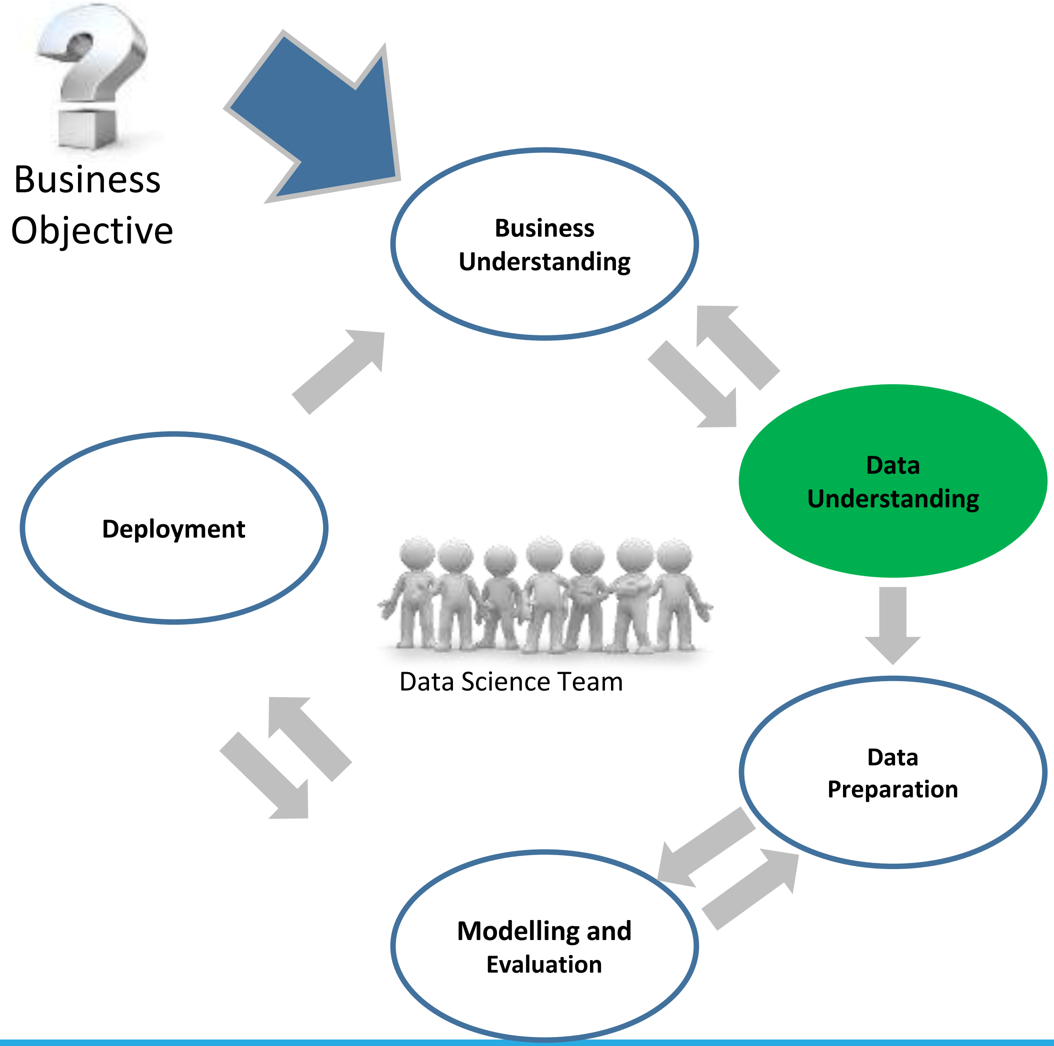
Data Science Conceptual Model – Crisp DM outline Methodology



Business Understanding

- IDENTIFYING YOUR BUSINESS GOALS
    - A problem that your management wants to address
    - The business goals
    - Constraints (limitations on what you may do, the kinds of solutions that can be used, when the work must be completed, and so on)
    - Impact (how the problem and possible solutions fit in with the business)
- ASSESSING YOUR SITUATION
    - Inventory of resources: A list of all resources available for the project.
    - Requirements, assumptions, and constraints:
    - Risks and contingencies:
    - Terminology
    - Costs and benefits:
- DEFINING YOUR DATA-MINING GOALS
    - Data-mining goals: Define data-mining deliverables, such as models, reports, presentations, and processed datasets.
    - Data-mining success criteria: Define the data-mining technical criteria necessary to support the business success criteria. Try to define these in quantitative terms (such as model accuracy or predictive improvement compared to an existing method).
- PRODUCING YOUR PROJECT PLAN
    - Project plan: Outline your step-by-step action plan for the project. (for example, modelling and evaluation usually call for several back-and-forth repetitions).
    - Initial assessment of tools and techniques

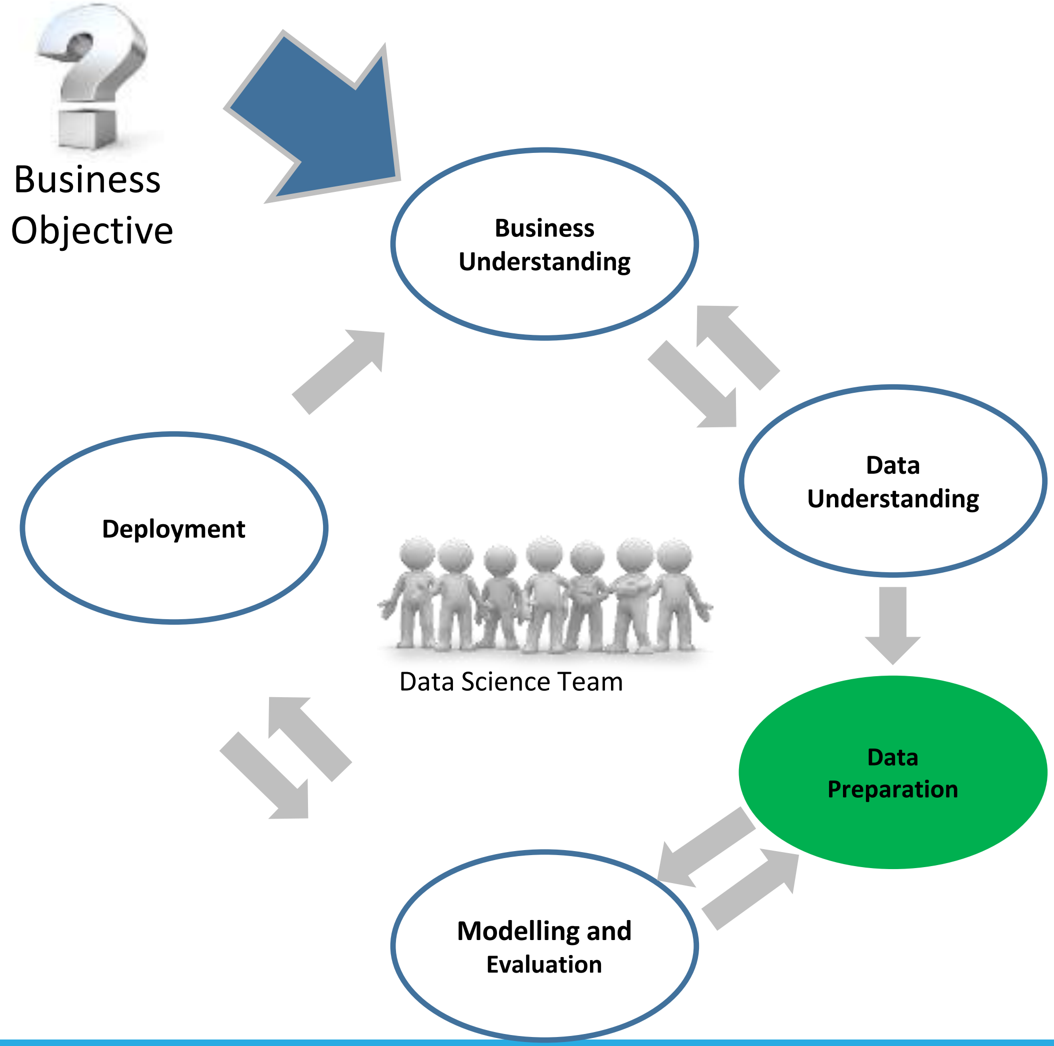
Data Science Conceptual Model – Crisp DM outline Methodology



Data Understanding

- GATHERING DATA
    - Outline data requirements: Create a list of the types of data necessary to address the data mining goals. Expand the list with details such as the required time range and data formats.
    - Verify data availability: Confirm that the required data exists, and that you can use it.
    - Define selection criteria: Identify the specific data sources (databases, files, documents, and so on.)
- DESCRIBING DATA
    - Now that you have data, prepare a general description of what you have.
- EXPLORING DATA
    - Get familiar with the data.
    - Spot signs of data quality problems.
    - Set the stage for data preparation steps.
- VERIFYING DATA QUALITY
    - The data you need doesn’t exist. (Did it never exist, or was it discarded? Can this data be collected and saved for future use?)
    - It exists, but you can’t have it. (Can this restriction be overcome?)
    - You find severe data quality issues (lots of missing or incorrect values that can’t be corrected).

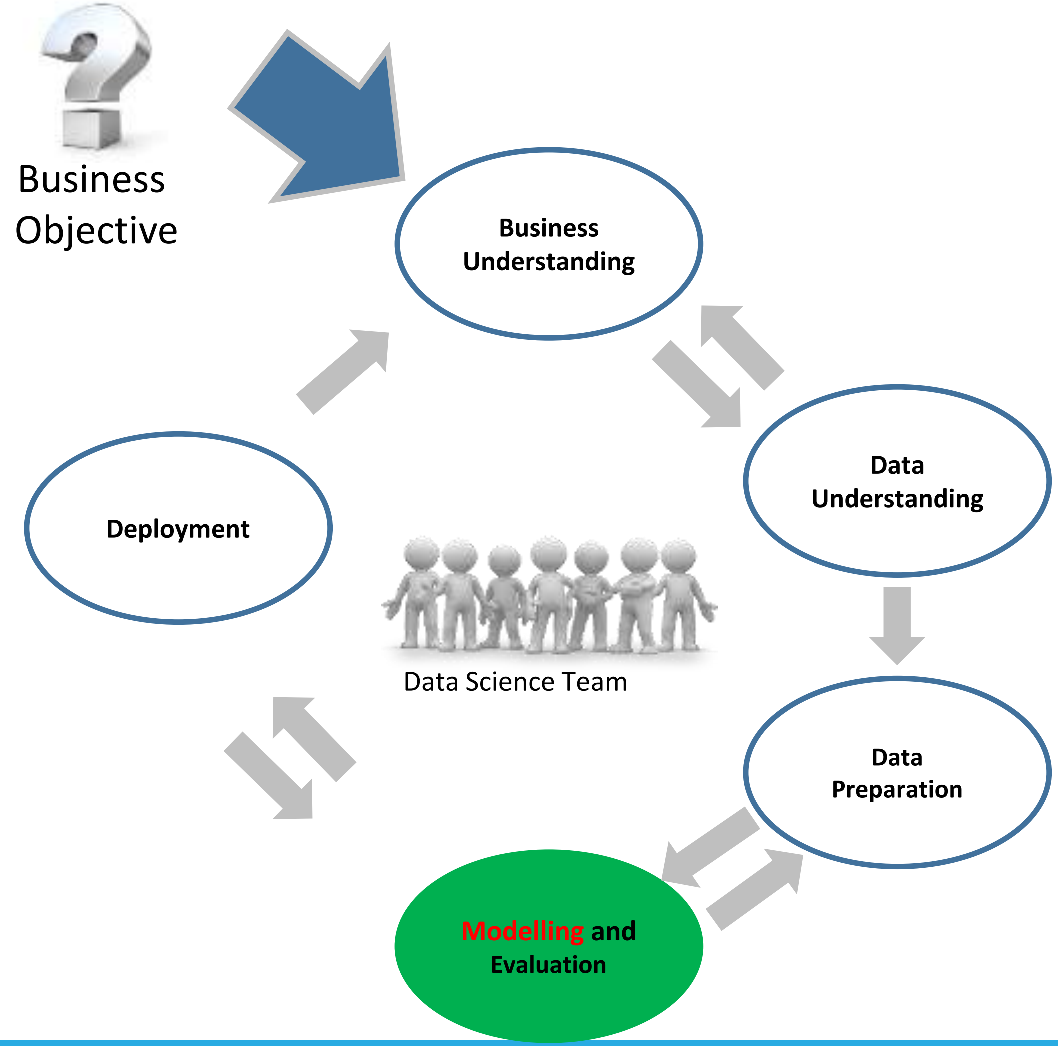
Data Science Conceptual Model – Crisp DM outline Methodology



Data Preparation

- SELECTING DATA
  - Now you will decide which portion of the data that you have is actually going to be used for data mining.
  - The deliverable for this task is the rationale for inclusion and exclusion. In it, you'll explain what data will, and will not, be used for further data-mining work.
  - You'll explain the reasons for including or excluding each part of the data that you have, based on relevance to your goals, data quality, and technical issues
- CLEANING DATA
  - The data that you've chosen to use is unlikely to be perfectly clean (error-free).
  - You'll make changes, perhaps tracking down sources to make specific data corrections, excluding some cases or individual cells (items of data), or replacing some items of data with default values or replacements selected by a more sophisticated modelling technique.
- CONSTRUCTING DATA
  - You may need to derive some new fields (for example, use the delivery date and the date when a customer placed an order to calculate how long the customer waited to receive an order), aggregate data, or otherwise create a new form of data.
- INTEGRATING DATA
  - Your data may now be in several disparate datasets. You'll need to merge some or all of those disparate datasets together to get ready for the modelling phase.
- FORMATTING DATA
  - Data often comes to you in formats other than the ones that are most convenient for modelling. (Format changes are usually driven by the design of your tools.) So convert those formats now.

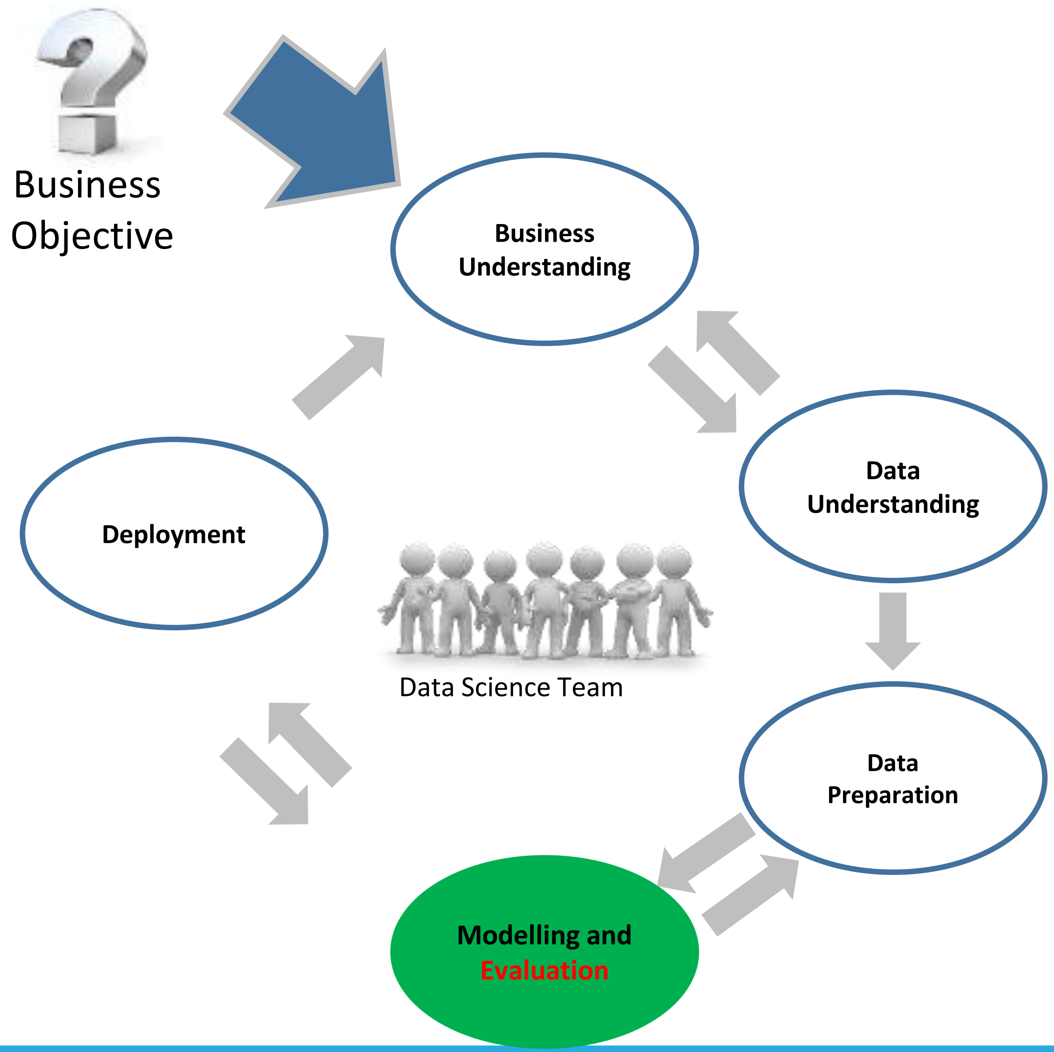
Data Science Conceptual Model – Crisp DM outline Methodology



Modelling and Evaluation (Modelling)

- SELECTING MODELING TECHNIQUES
  - Modelling technique: Specify the technique(s) that you will use.
  - Modelling assumptions: Many modelling techniques are based on certain assumptions.
- DESIGNING TESTS
  - The test in this task is the test that you'll use to determine how well your model works. It may be as simple as splitting your data into a group of cases for model training and another group for model testing.
  - Training data is used to fit mathematical forms to the data model, and test data is used during the model-training process to avoid overfitting
- BUILDING MODEL(S)
  - Parameter settings: When building models, most tools give you the option of adjusting a variety of settings, and these settings have an impact on the structure of the final model. Document these settings in a report.
  - Model descriptions: Describe your models. State the type of model (such as linear regression or neural network) and the variables used.
  - Models: This deliverable is the models themselves. Some model types can be easily defined with a simple equation; others are far too complex and must be transmitted in a more sophisticated format.
- ASSESSING MODEL(S)
  - Model assessment: Summarizes the information developed in your model review. If you have created several models, you may rank them based on your assessment of their value for a specific application.
  - Revised parameter settings: You may choose to fine-tune settings that were used to build the model and conduct another round of modelling and try to improve your results.

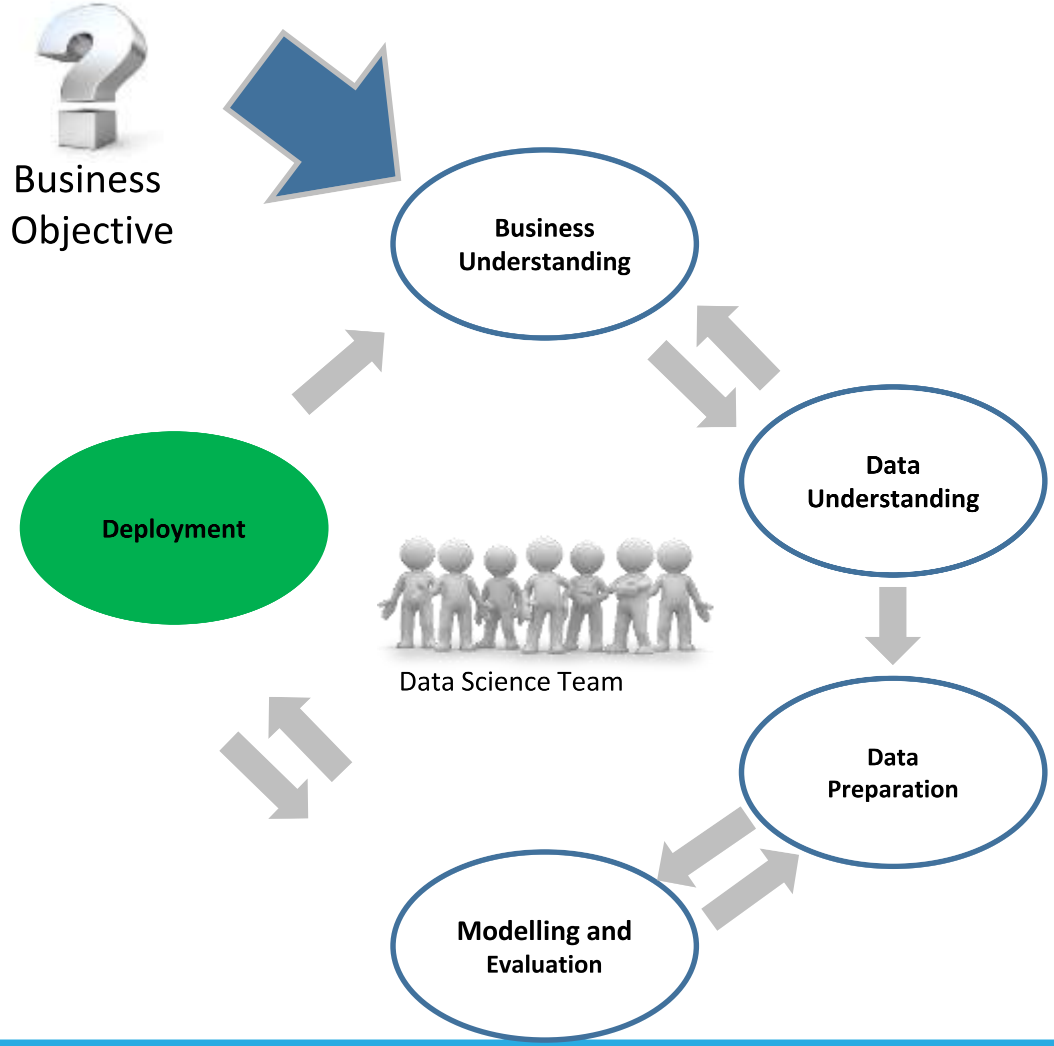
Data Science Conceptual Model – Crisp DM outline Methodology



Modelling and Evaluation Cont… (Evaluation)

- EVALUATING RESULTS
  - Assessment of results (for business goals): Summarize the results with respect to the business success criteria that you established in the business-understanding phase. Explicitly state whether you have reached the business goals defined at the start of the project.
  - Approved models: These include any models that meet the business success criteria.
- REVIEWING THE PROCESS
  - Now that you have explored data and developed models, take time to review your process. This is an opportunity to spot issues that you might have overlooked and that might draw your attention to flaws in the work that you've done while you still have time to correct the problem before deployment. Also consider ways that you might improve your process for future projects.
- DETERMINING THE NEXT STEPS
  - List of possible actions: Describe each alternative action, along with the strongest reasons for and against it.
  - Decision: State the final decision on each possible action, along with the reasoning behind the decision.

Data Science Conceptual Model – Crisp DM outline Methodology



Deployment

- PLANNING DEPLOYMENT
  - When your model is ready to use, you will need a strategy for putting it to work in your business.
- PLANNING MONITORING AND MAINTENANCE
  - Data-mining work is a cycle, so expect to stay actively involved with your models as they are integrated into everyday use.
- REPORTING FINAL RESULTS
  - Final report: The final report summarizes the entire project by assembling all the reports created up to this point, and adding an overview summarizing the entire project and its results.
  - Final presentation: A summary of the final report is presented in a meeting with management. This is also an opportunity to address any open questions.
- REVIEW PROJECT
  - Finally, the data-mining team meets to discuss what worked and what didn't, what would be good to do again, and what should be avoided!

Data Science Continuous Integration and improvement Cycle