**2020/01/18**

**MODELLING METHODOLOGIES**

**Sebastien Perez Vasseur**

Las empresas ya tienen soluciones para sus problemas de negocio.

Pero el mundo ha cambiado. Ahora existe:

1. **Cloud** no es una solución tecnológica, es una solución de marketing. Máquina, software, mantenimiento: lo hace la empresa de cloud y tú pagas por su uso. Así siempre dispones de la mejor máquina, software actualizado, etc.
2. **Data**: se produce más y se publican más (páginas del gobierno, etc.).
3. **Machine** **Learning**: hay muchas librerías open source disponibles.
4. **IoT and sensor:** ayudan a monitorizar, obtener datos y optimizar tareas.

La gran pregunta sería : ¿cómo hacer para que una empresa resuelva sus problemas usando machine learning?

En resumen, el aumento de los datos disponibles, el acceso a mejor hardware y herramientas más eficientes y open source, dentro de un entorno cloud equivale a OPORTUNIDADES.

**¿Qué es Machine Learning?**

De la misma manera que tareas físicas sencillas se automatizaron en los 70s con el uso de Robots, Machine learning nos permite automatizar tareas cognitivas sencillas. Las tareas deben ser sencillas y ensambladas para producir resultados. Aunque a veces, demasiada automatización puede conducir a mala calidad del resultado, como el caso de Tesla y su línea de producción 100% automatizada. Se necesitan personas controlando las atuomatizaciones.

**Types of learning**

**1. Generalization & relation (Supervised learning):**

- Learning from *past* elements.

- De cada elemento se recopilan features o características.

- A cada elemento se le asignan etiquetas (manualmente).

- Buscar relación entre las características y el target según la etiqueta que tenga habrá dos tipos:

- Regression: si el target es numérico (continuous value output). Predict a number.

- Classification: si el target es categórico (discrete value output). Predict a label.



**2. Comparison (Unsupervised learning):**

- Learning by comparison.

- Extraer características de los elementos de manera automática.

- Definir distancia entre elementos a partir de las características.

- Clustering: Con esa distancia creo grupos/segmentos. Finding elements alike.

- Dimensionality reduction: reducir número de caracterísitcas de los elementos. Explaining elements with less attributes.

**3. Reinforcement (Reinforcement Learning): no lo veremos en el master**

Learn to do something yourself purely by maximasing your expected reward.

**NOTAS**:

- EL 99% de los problemas de los negocios actualmente se resuelven con **Generalización** o **Comparación**.

**- Supervised learning**, le llaman supervisado por que tiene etiquetas que tienen que ser asignada manualmente.

- **AI** es un campo mucho más amplio que el machine learning.

- **Deep Learning** es un tipo de modelo de machine learning para encontrar relaciones

- **Statistics**: es un campo de las Matemáticas. Encuentra relaciones de manera matemática. Machine Learning utiliza estadística pero también fuerza bruta.

**What is ML about?**

**PROBLEM STATEMENT: Predecir precio de una casa acorde a sus características.**

Coger elementos del pasado y etiquetar. El target sería el precio de la casa.

¿Qué características de la casa escojo? Ese es el curro de verdad, descubrir qué características ayudarán a definir el precio de la casa.

Un modelo es la manera de encontrar la relación entre las características y el target.

**Machine Learning is about features**

La realidad se explica mediante una tabla de características.

**Machine Learning is about models**

Se usan modelos para explicar las relaciones entre esas características. En el ejemplo de las casas, la función que buscamos se expresaría así:

SalePrice = f(Features)

**Tipos de modelos**

Se pueden usar muchos modelos para un tarea cognitiva. El Data Scientist debe elegir el que mejor se comporte para esa tarea.

**- Linear Regression**: tries to find a linear function. Este es un modelo matemático pero los demás de abajo no! son modelos *computacionales*.

**- Decision Tree**: coge una característica, si su valor es más de tal ve a esta rama, si no ve a esta otra y así recursivamente. Hay que encontrar los mejores cortes, o ramas para ver la relación de las características con el precio. Needs to find the best splits for the prediction.

**- Random forest.**

**- K-neightbours.**

**- SVM: Maquina de Vector de Soportes.**

**- Neural Networks.**

**- Deep learning.**

**- And more….**

Como hay muchos modelos puedes probar todos y ver cuál te da más grados de acierto.

**NOTAS**:

Lo modelos matemáticos abstraen los millones de datos en formulas sencillas. Los modelos computacionales guardan esos millones de datos en realidad.

**Parameters of the model**

El Data Scientist trabaja por encontra los mejores parámetros. Cuáles son estos parámetros?

Parámetros NO son las características elegidas. El modelo normalmente elige los mejores parámetros por ti.

Linear Regression: a, b, c y d.

Decision Tree: number of splits.

**Training**

Mediante entrenamiento se encuentra los mejores parámetros. El entrenamiento es un problema de búsqueda porque estás buscando los mejores parámetros.

Un modelo está entrenado cuando ha encontrado los mejores parámetros. Al conjunto de datos empleados para entrenar el modelo se los llama Training Data.

**¿Cómo se evalúa un modelo? Métrics**

Cojo un casa cuyo precio es conocido y lo meto en el modelo a ver cómo predice su precio. Usas varios modelos y ves cuál lo predice mejor. Uso métricas para medir la diferencia de valores que me ha dado un modelo u otro. Algunas métricas son:

* Bias: average of errors.
* MAE (Mean Absolute Error): average of the absolute value of errors.
* RMSE (Root-Mean-Square Error): square root of average of the square of errors…

La métrica al final la define el negocio.

**Evaluate Models: Training – Test Split**

Los datos disponibles se dividen en dos:

- Uno para calcular parámetros: **Training Set.**

- Uno para calcular las métricas: **Test Set.**

Ejemplo, si tengo 1 millón de casas, uso 900000 (training set) para entrena y 100000 para testear el modelo (test set). El test sirve para comprar el resultado del modelo con el dato real conocido y calcular las métricas.

**Final Model**

Una vez que se tienen los mejores parámetros para un tipo de modelo, se escoge el modelo con la mejor métrica, y ese es el que uso para predecir el precio de la casa.

Ejemplo:

- Linear Regression – MAE: 15.

- Decision Tree – MAE: 10.

**SUMMARY:**

Machine learning is about **features**, **models** and **metrics**.:

- ML requires explaining reality as a table of **features**.

- We use **models** to explain relationships between features for the given ML task.

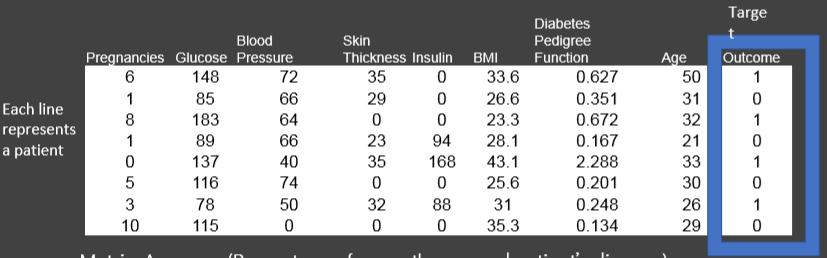
- Each model is evaluated according to a **metric**.

**ML EXAMPLES:**

**Classification:**

**Problem**: predecir diabetes en pacientes. Es un ejemplo en que el resultado es o 1 o 0, o tienes diabetes o no.

**Features**:

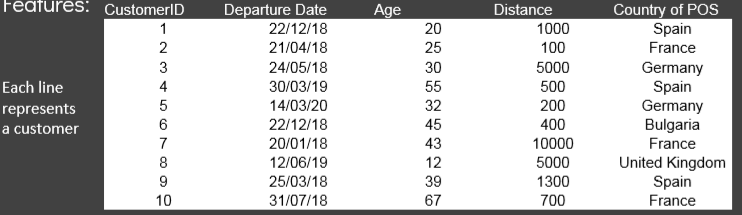


**Metric**: Accuracy (porcentaje de diagnósticos correctos).

**Segmentation:**

**Problem**: detectar grupos de clientes similares. No tengo etiquetas, tengo características. La segmentación está basada en buscar grupos.

**Features**:



**Method:** find groups of simliar customers based on their proximity.

**Metric**: Silhouette index, dice cuán lejos está un consumidor de otro cluster que no es el suyo.

**Ejemplo - Panadero:**

Problema, vender todas las barras en un día para que no le sobre.

¿Qué queremos predecir? número de clientes.

¿Qué tipo de problema tengo? numérico, escojo **regresión**.

**Ejemplo - Operaria de Fábrica:**

Problema: que las máquinas nunca se paren porque se hayan estropeado.

Pregunta a predecir: ¿sí o no se estropeará una maquina mañana? es un problema de **clasificación**.

**Ejemplo – Responsable de Marketing:**

Problema: se me van los clientes.

¿Qué queremos predecir? conversión de clientes. Es un problema de clasificación.

**Ejemplo – Médico Radiólogo:**

Problema: dar un mal diagnóstico.

¿Qué queremos predecir? sí o no hay enfermedad en la radiografía. Es un problema de clasificación.

**ML CANVAS: Translating a Business Problem into Machine Learning**

Cuando hay un problema de negocio tengo que descomponerlo para intentar resolverlo. Será necesario hablar de:

- Data Sources

- History

- Features

- Frecuency

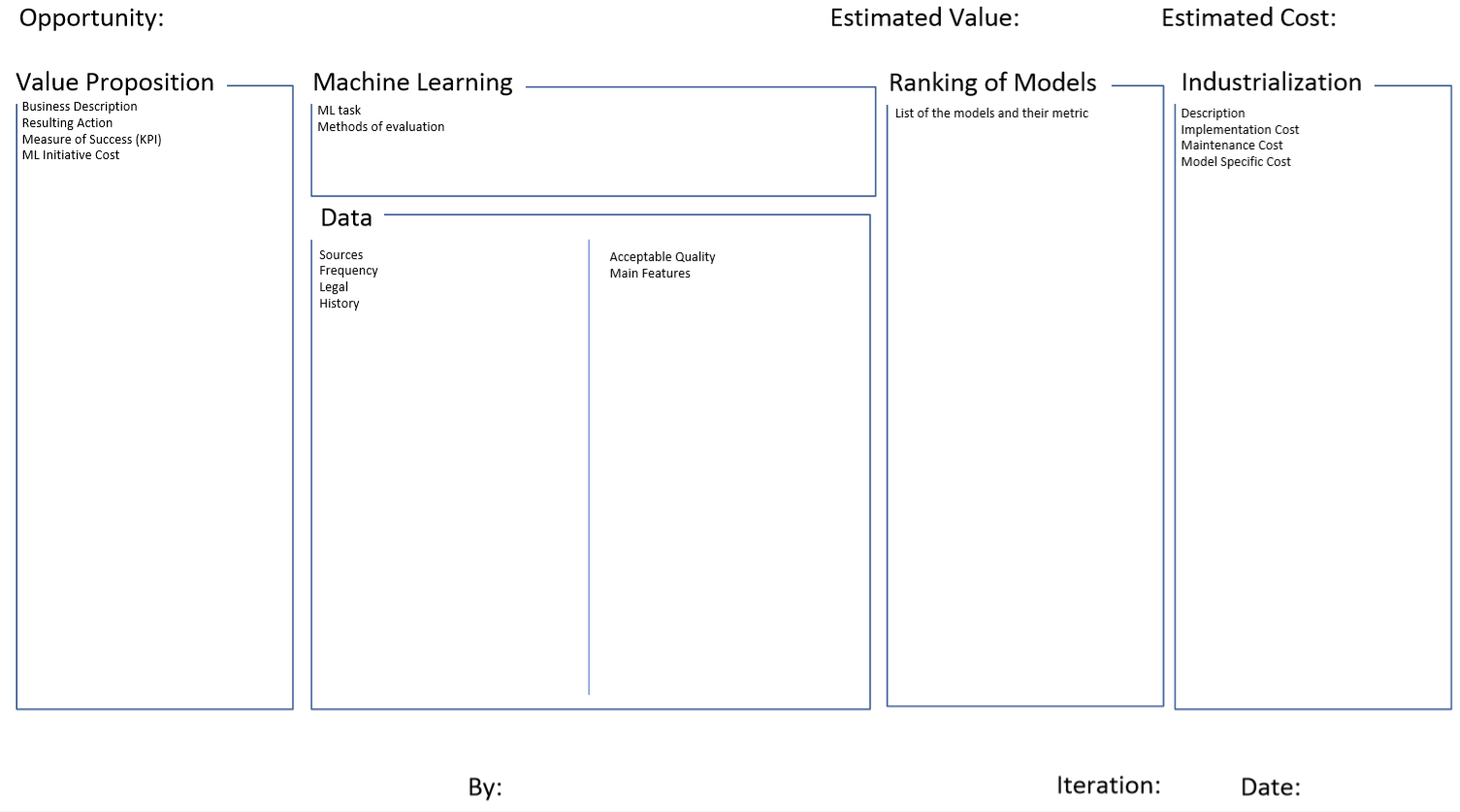
- Quality

- Volume

**- Evaluation of success:** normalmente la persona de business me dirá cuál es el valor y la medida de éxito

**- How does it translate to a metric**

Para acometer todos estos puntos se realiza un ML Canvas:



**Data Science Process**

1. **Value proposition**

2. **Data**: extraction of features.

Data Sources need to be completed in a small table with:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Frecuency | Legal | History |
|  | Cada cuanto se actualizan | Son públicos o privados |  |

3. **Machine Learning**: selection of ML task and evaluation (metric).

The ML table is created through a data flow with basic steps shown below:

[**4.1   Basic Stages**](https://pypi.org/project/pdpipe/#id17)

*AdHocStage* - Define custom pipeline stages on the fly.

*ColDrop* - Drop columns by name.

*ValDrop* - Drop rows by their value in specific or all columns.

*ValKeep* - Keep rows by their value in specific or all columns.

*ColRename* - Rename columns.

*DropNa* - Drop null values. Supports all parameter supported by pandas.dropna function.

*FreqDrop* - Drop rows by value frequency threshold on a specific column

*ColReorder* - Reorder columns.

*RowDrop* - Drop rows by callable conditions.

[**4.2   Column Generation**](https://pypi.org/project/pdpipe/#id18)

*Bin* - Convert a continuous valued column to categoric data using binning.

*OneHotEncode* - Convert a categorical column to the several binary columns corresponding to it.

*MapColVals* - Replace column values by a map.

*ApplyToRows* - Generate columns by applying a function to each row.

*ApplyByCols* - Generate columns by applying an element-wise function to columns.

*ColByFrameFunc* - Add a column by applying a dataframe-wide function.

*AggByCols* - Generate columns by applying a series-wise function to columns.

*Log* - Log-transform numeric data, possibly shifting data before.

[**4.3   Scikit-learn-dependent Stages**](https://pypi.org/project/pdpipe/#id19)

*Encode* - Encode a categorical column to corresponding number values.

*Scale* - Scale data with any of the sklearn scalers.

[**4.4   nltk-dependent Stages**](https://pypi.org/project/pdpipe/#id20)

*TokenizeWords* - Tokenize a sentence into a list of tokens by whitespaces.

*UntokenizeWords* - Joins token lists into whitespace-seperated strings.

*RemoveStopwords* - Remove stopwords from a tokenized list.

*SnowballStem* - Stems tokens in a list using the Snowball stemmer.

*DropRareTokens* - Drop rare tokens from token lists.

4. **Ranking of models**: model creation, quantity of data > parameters, loop until satisfied.

¿Qué pasa si el mejor de tus modelos no es satisfactorio?

- Meter más datos, líneas a tu tabla.

- Meter nuevas características.

- Quitar características: porque puede haber características que empeoren los resultados. Una cosa que se hace para ver qué características no son importantes es intercambiar el contenido de dos columnas y ejecutar el modelo, si el resultado no cambia mucho es que esas características no son importantes.

- Probar nuevos modelos.

*Iterate between 2 and 4 until satisfied.*

5. **Industrialization**: integrar el modelo creado en la parte de negocio. ¿Cómo se usará el modelo? ¿Cuándo (a menudo) se usará el modelo? ¿Cada cuánto se actualizará (re-entrenará) el modelo? ¿Qué decisión se toma cuando se usa el modelo?

Deliverables: El data scientist entrega un modelo. O a veces simplemente dice qué modelo/s es mejor usar para este problema.

**NOTAS**:

Es necesario saber interpretar/explicar los modelos. Hay modelos fáciles de explicar como Linear Regression o Decision Tree. Pero otros que son difíciles o imposibles de explicar a un cliente cómo han llegado a tal o cual resultado.

Seguros y Banca no permiten por ley usar modelos difíciles de explicar. Por ejemplo, no pueden usar Deep Learning.

HANDS ON EXERCISES:

- Weather Forecast.

- Moview Recommendation.

- Email Spam Filter.

- Pollution Prediction.

- Customer Segmentation.