

**Manoel Fernando Alonso Gadi**

Professor Asociado

[manoel.gadi@uah.es](mailto:manoel.gadi@uah.es)



Universidad  
de Alcalá

# **CONCEPTOS DE BIG DATA, BUSINESS ANALYTICS Y BUSINESS INTELLIGENCE**



---

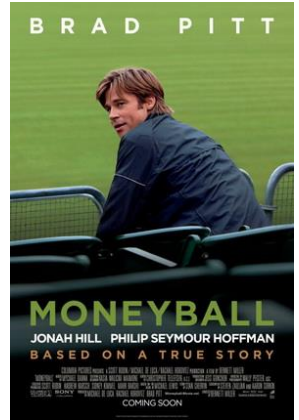
## DO YOU FEAR AN ANALYTICS DRIVEN COMPETITOR? YOU SHOULD!

AREA: IE SCHOOL OF SOCIAL AND BEHAVIORAL SCIENCES

PROFESSOR: **MANOEL F. A. GADI**

E-mail: [mfalonso@faculty.ie.edu](mailto:mfalonso@faculty.ie.edu)

Disruptive business models and disruptive technology are generating a frenzy in the race to have the best of breed technology and systems. The same is becoming reality in the field of statistical & computational models for Analytics. Companies like Capital One with its thousand simultaneous tests, Amazon with its recommendation systems and Google with its raking algorithm among others formed an army which took Analytics to a complete new level by making prescriptive modeling and testing the bedrock of its competitive advantage. In the

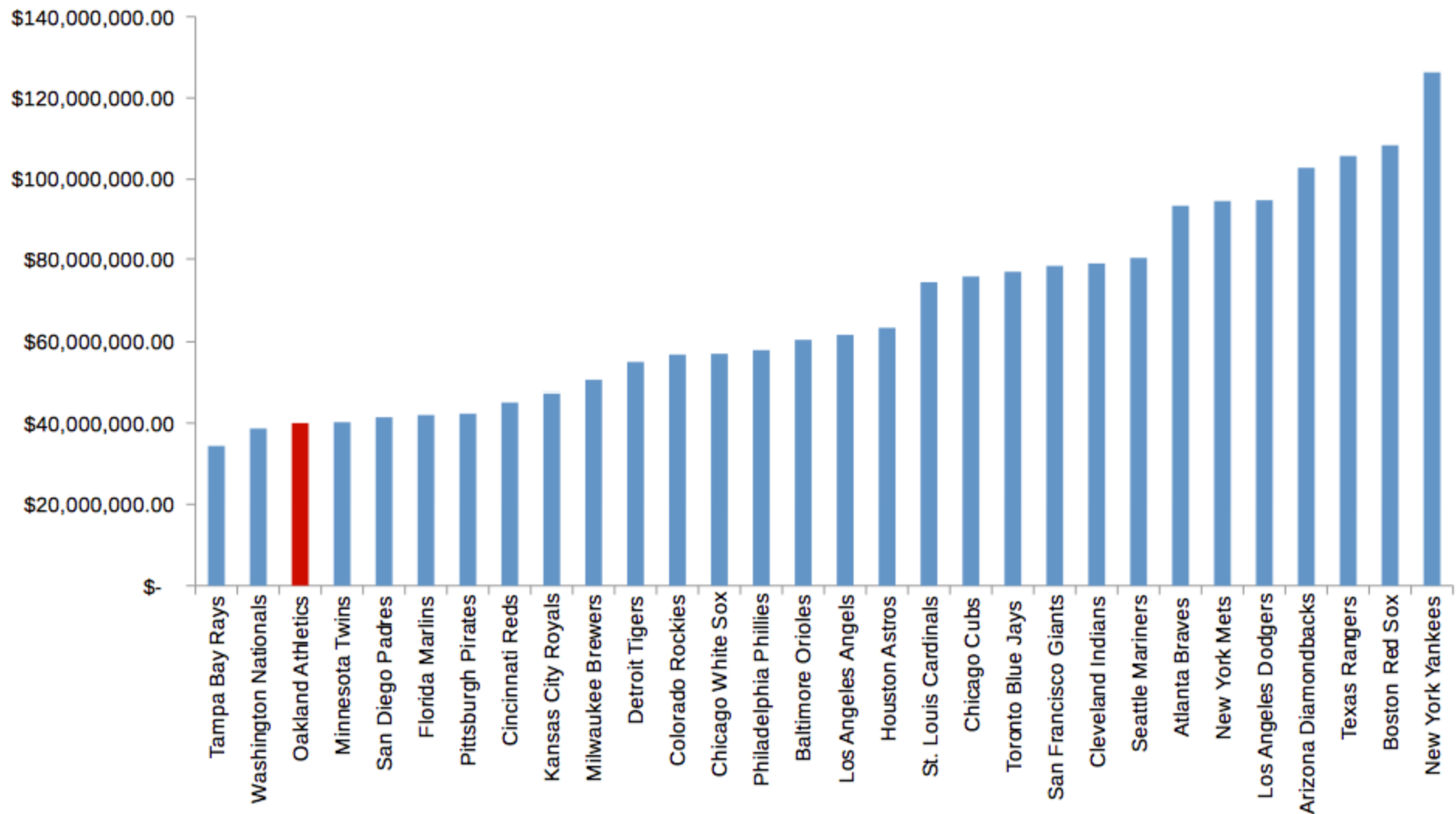


➤ Moneyball -

<http://www.youtube.com/watch?v=AiAHLZVgXjk>

- Which type of company is Oakland Athletics?

# Moneyball Year (2002) MLB Team Salaries



## *Machine Learning - Model estimation*

$$f(X) = y$$

# Machine Learning - Model estimation

$$f(X) = y$$

ib_var_1	ib_var_2	ib_var_3	icn_var_22	icn_var_23	icn_var_24	ico_var_25	ico_var_26	ico_var_27	if_var_65	if_var_68	if_var_80	ob_target
0	1	0	2	1	2	4	4	4	31	4626	0.7666667	0
0	1	0	3	1	1	5	5	4	34	822.4	0.7	0
0	0	0	4	1	1	4	4	4	50	1028	0.6666667	0
0	1	0	2	1	3	4	4	4	30	4112	0.7666667	0
0	0	0	3	1	1	4	4	4	37	1028	0.7333333	0
0	1	0	2	1	1	5	5	5	34	1543.5	0.6	0
0	1	0	3	1	3	5	5	3	27	5140	0.8666667	0
0	0	0	4	1	1	5	4	4	27	1542	0.7333333	0
0	1	0	2	5	3	4	3	5	39	514	0.5333333	0
0	0	0	3	1	1	4	4	4	39	1542	0.6	0
0	1	0	3	1	2	4	5	4	39	6168	0.8	0
0	0	0	1	1	1	4	4	4	40	1028	0.6666667	0
0	1	0	2	1	1	4	5	3	37	2056	0.7666667	0
0	1	0	2	1	1	4	4	4	30	6168	0.7	1
0	0	0	2	1	1	5	5	5	52	514	0.8	0
1	0	0	2	1	1	4	4	4	48	205.6	0.8	1
0	0	0	3	1	2	5	5	5	38	1028	0.6	1
0	1	0	2	1	3	4	4	3	26	805.6	0.7	0
0	0	0	2	1	1	4	4	4	57	1542	0.7666667	0
0	1	0	2	1	2	4	3	4	43	3598	0.8	0
0	1	0	2	1	1	4	4	4	39	1028	0.6	0
0	1	0	2	1	1	5	5	5	27	2056	0.6	0
1	1	0	2	1	2	5	5	5	40	1028	0.8	0
0	1	0	2	1	1	4	4	4	42	925.2	0.7333333	0
0	0	0	3	3	1	3	4	5	55	3084	0.6	1
0	1	0	2	1	1	4	5	4	45	616.8	0.7666667	0

$$f(X) = y$$

**X = data frame with observed characteristics and data points.**

**y = target variable, variable we wish to predict**

# *Machine Learning - Example*

**Country: Spain**

**Company: Banco Santander**

**Task: Collection and Recovery activities**

**Machine Learning: supervised – classification**



$$f(X) = y$$

## JUEGO DE ROLES – Vuelta 1

Pensar en una empresa (nombra el país y el sector y el problema) que aplica el aprendizaje automático a su negocio. En su participación, siendo muy breve (no más de 280 caracteres), informe el país, sector de la empresa, el nombre de la empresa y explique cuál es el problema que intenta predecir o solucionar, por favor, en su explicación, mencione si utiliza el método Supervisado o No supervisado y, si es supervisado, mencione si es un problema de clasificación o regresión.

---- Ejemplo:

País: España

Empresa: Banco Santander

Tarea: Actividades de recolección y recuperación

Aprendizaje automático: supervisado – clasificación

<https://docs.google.com/spreadsheets/d/1O2-m0QD0j4pPXhtOyafH2OezMQ8kSO9D6w-qo1xfgMQ/edit?usp=sharing>

**X:**

- **Population:**
- **Characteristics:**

**y:**

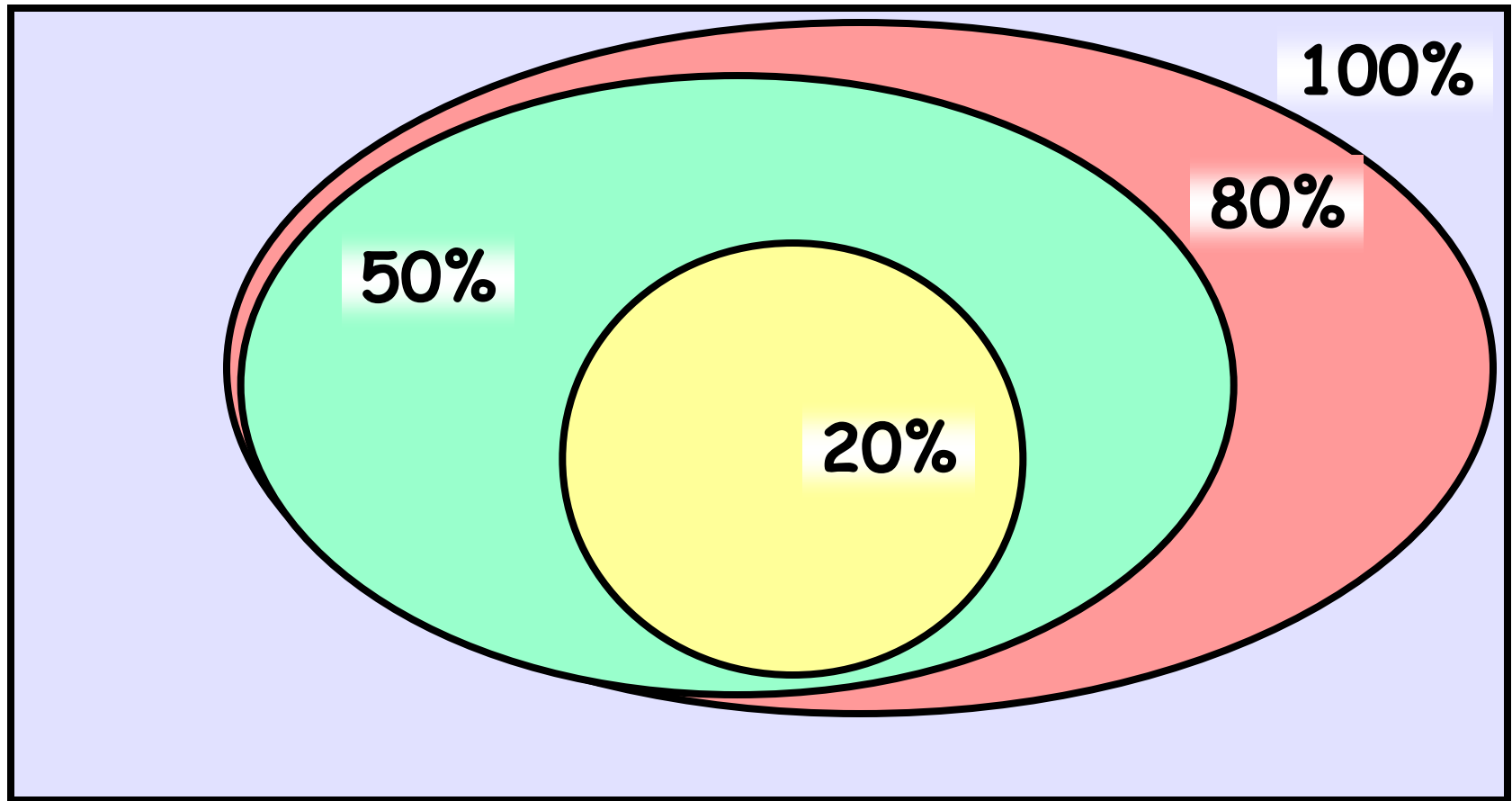
- **Target definition:**

Every Human

Living in Spain

Existing Customers

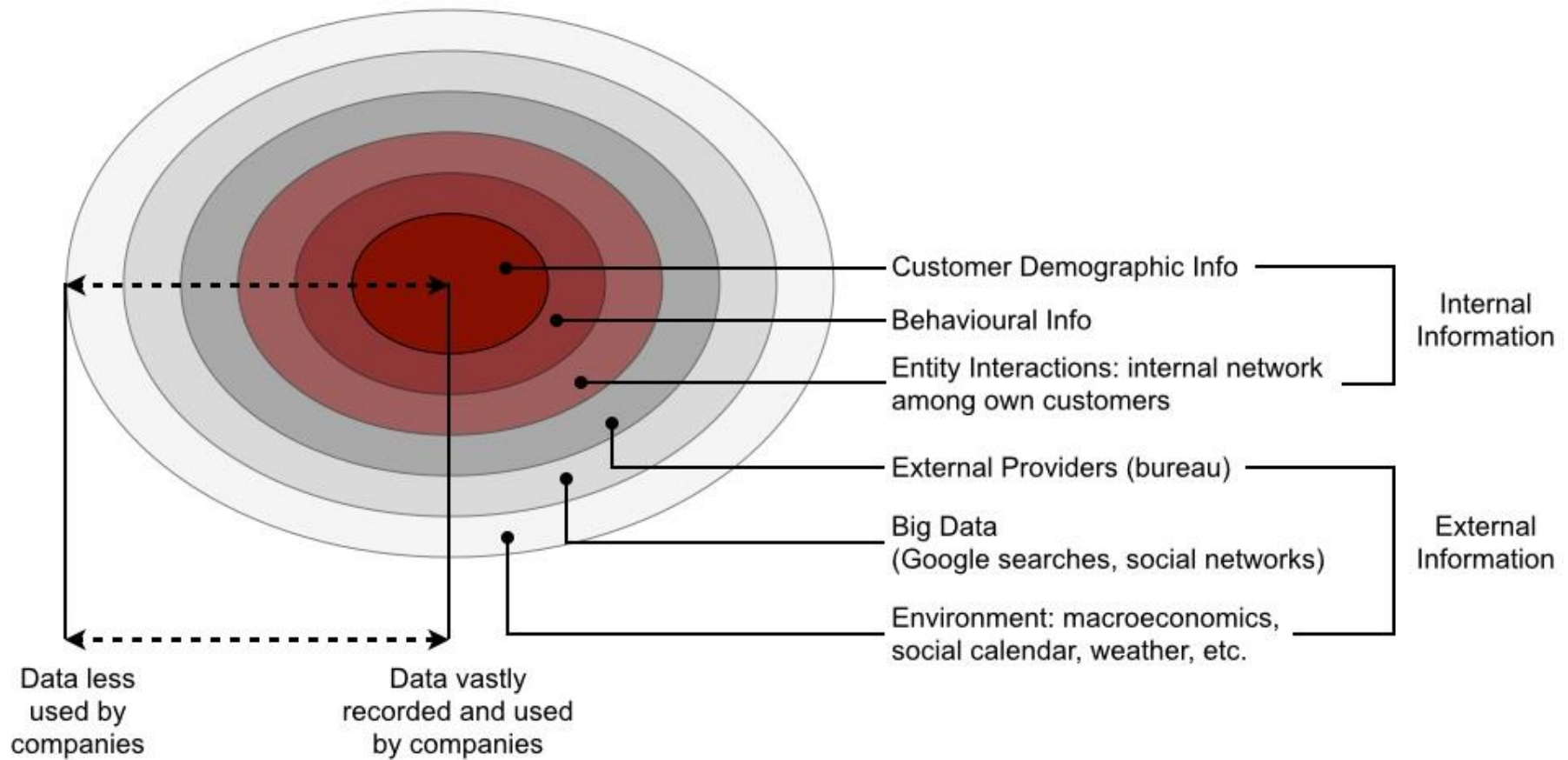
Default (Unpaid Debt)



Population: Existing Customers in Spain with Unpaid Debts.

$$f(\text{X}) = y$$

## Map of information available for businesses



# *Machine Learning - Model estimation*

$$f(X) = y$$

**X:**

- **Population:**

Existing Customers in Spain with Unpaid Debts

- **Characteristics:**

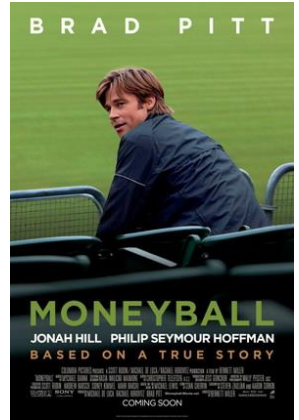
- Time on books
- Total Paid last 12 months
- Total Debt in Arrears

**y:**

- **Target definition:**

1 – good – customer who repaid more than 90% of the debt in arrears the following 6 months counting from the payment due date.

0 – bad – every one else.



➤ Moneyball - <http://www.youtube.com/watch?v=AiAHlZVgXjk>

➤ Moneyball (Breaking Biases) -  
<http://www.youtube.com/watch?v=yGf6LNWY9AI>

$$f(X) = y$$

## JUEGO DE ROLES – Vuelta 2

Agregue detalles a su propia respuesta de la ronda anterior, agregando la siguiente información:

- Definición de población
- 3 ejemplos de características
- Definición de la variable independiente (target)

Si crees que no tienes toda la información, responde cómo crees que deberían ser las respuestas (¡puedes inventar!), por favor no dejes la respuesta en blanco.

--- Ejemplo de Respuesta:

X:

- Población: Clientes existentes en España con deudas impagas
- Características:
  - Tiempo como cliente
  - Total pagado en los últimos 12 meses
  - Deuda total en mora

y:

- Definición de la variable target:
  - 1 - bueno - cliente que pagó más del 90% de la deuda en mora dentro de los 6 meses siguientes contados a partir de la fecha de vencimiento del pago.
  - 0 - malo - todos los demás.

DESCANSO

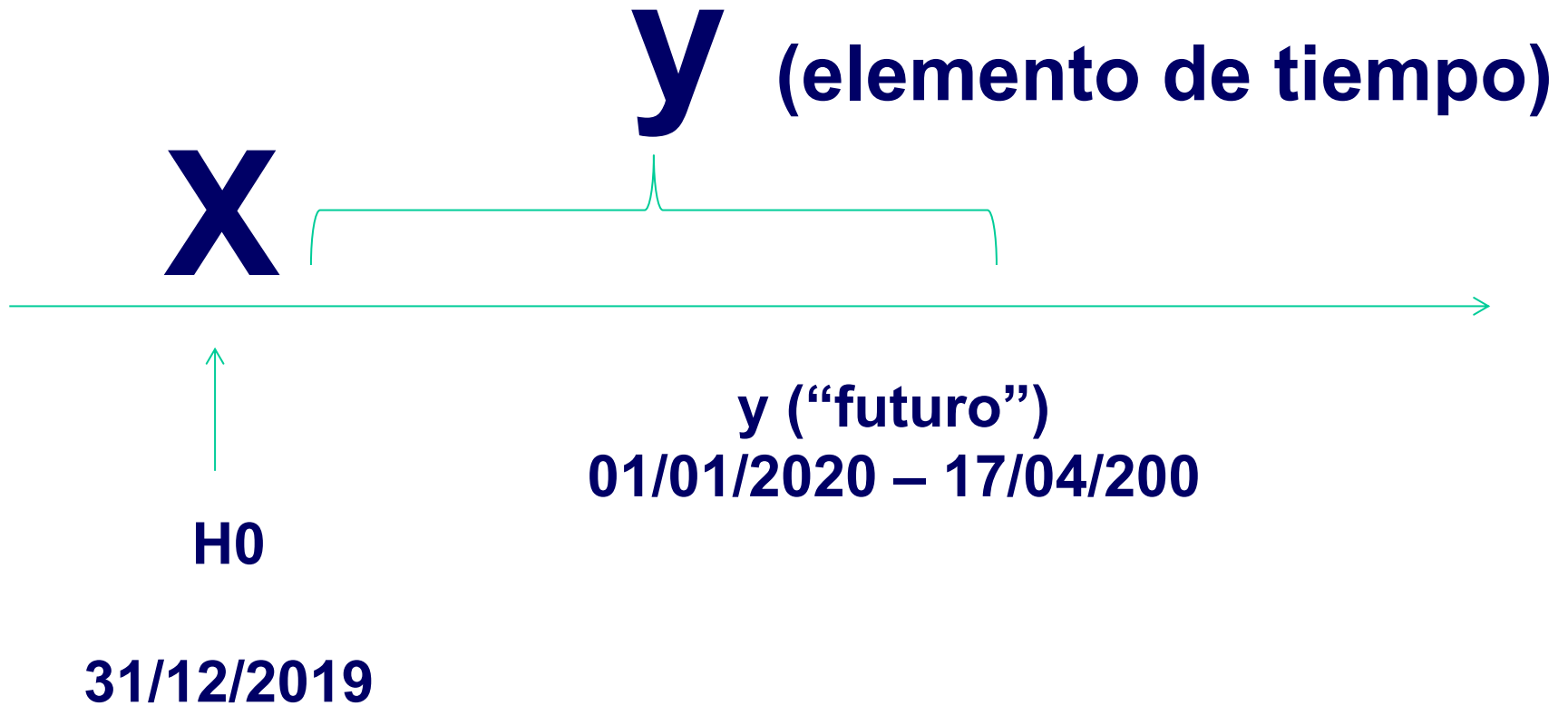
NOS VEMOS EN 20 MINUTOS!



$$f(X) = y$$

## Integrated vision of the Portfolio Management approach





# Machine Learning - Model estimation

$$f(X) = y$$

Characteristics (select)

Target  
definition  
(otro  
select)

Population (where)

$$f(X) = y$$

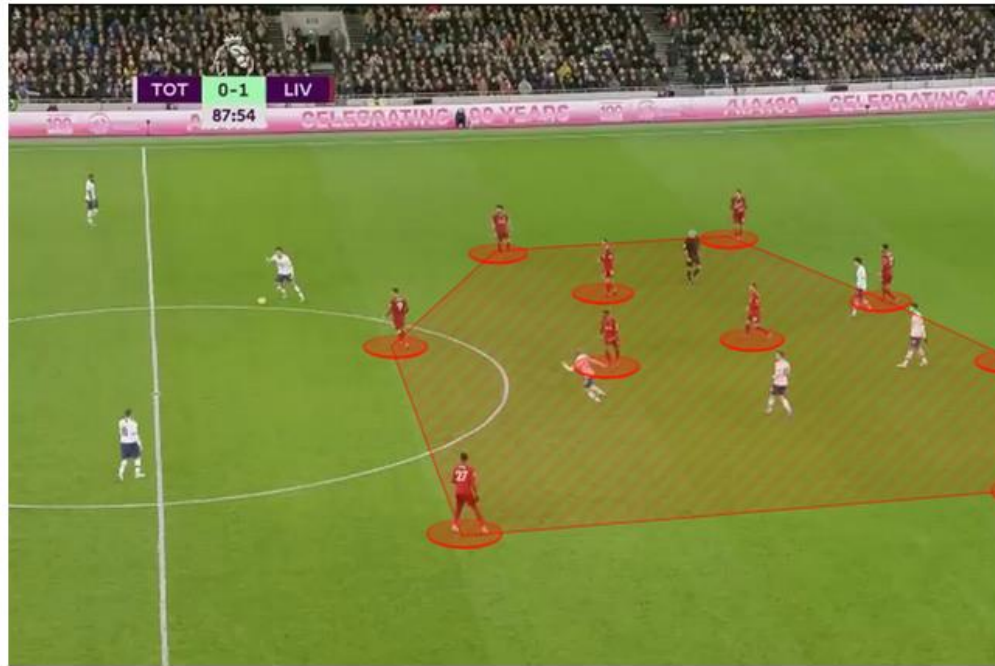
ib_var_1	ib_var_2	ib_var_3	icn_var_22	icn_var_23	icn_var_24	ico_var_25	ico_var_26	ico_var_27	if_var_65	if_var_68	if_var_80	cb_target
0	1	0	2	1	2	4	4	4	31	4626	0.7666667	0
0	1	0	3	1	1	5	5	4	34	822.4	0.7	0
0	0	0	4	1	1	4	4	4	50	1028	0.6666667	0
0	1	0	2	1	3	4	4	4	30	4112	0.7666667	0
0	0	0	3	1	1	4	4	4	37	1028	0.7333333	0
0	1	0	2	1	1	5	5	5	34	1543.5	0.6	0
0	1	0	3	1	3	5	5	3	27	5140	0.8666667	0
0	0	0	4	1	1	5	4	4	27	1542	0.7333333	0
0	1	0	2	5	3	4	3	5	39	514	0.5333334	0
0	0	0	3	1	1	4	4	4	39	1542	0.6	0
0	1	0	3	1	2	4	5	4	39	6168	0.8	0
0	0	0	1	1	1	4	4	4	40	1028	0.6666667	0
0	1	0	2	1	1	4	5	3	37	2056	0.7666667	0
0	1	0	2	1	1	4	4	4	30	6168	0.7	1
0	0	0	2	1	1	5	5	5	52	514	0.8	0
1	0	0	2	1	1	4	4	4	48	205.6	0.8	1
0	0	0	3	1	2	5	5	5	38	1028	0.6	1
0	1	0	2	1	3	4	4	3	26	805.6	0.7	0
0	0	0	2	1	1	4	4	4	57	1542	0.7666667	0
0	1	0	2	1	2	4	3	4	43	3598	0.8	0
0	1	0	2	1	1	4	4	4	39	1028	0.6	0
0	1	0	2	1	1	5	5	5	27	2056	0.6	0
1	1	0	2	1	2	5	5	5	40	1028	0.8	0
0	1	0	2	1	1	4	4	4	42	925.2	0.7333333	0
0	0	0	3	3	1	3	4	5	55	3084	0.6	1
0	1	0	2	1	1	4	5	4	45	616.8	0.7666667	0

## *Ejemplo Fútbol*



Jurgen Klopp

# Liverpool are using incredible data science during matches, and effects are extraordinary

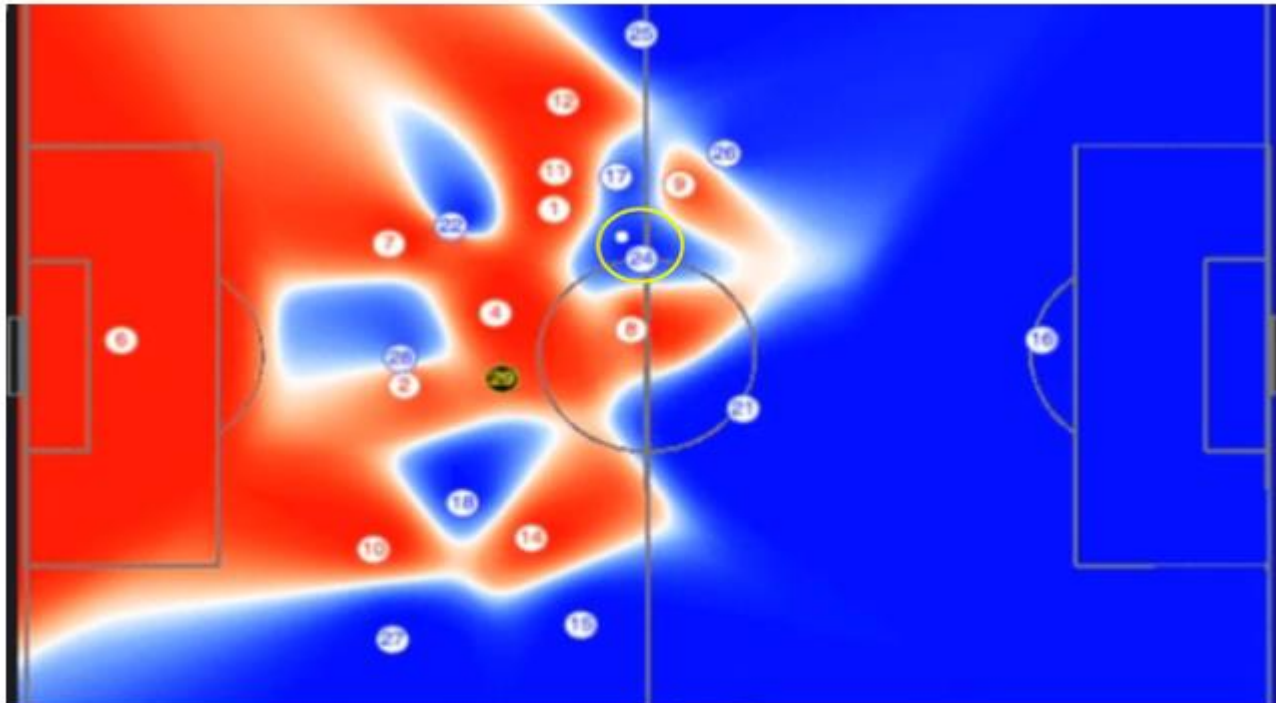


Liverpool form a compact block in the centre of the pitch late in the match with Spurs

source:

<https://www.liverpool.com/liverpool-fc-news/features/liverpool-transfer-news-jurgen-klopp-17569689>

# Liverpool are using incredible data science during matches, and effects are extraordinary



A pitch control visualisation which captures the regions of space controlled by certain players

The player who is circled in yellow has possession of the ball and, essentially, his team has access to the areas of the pitch that are blue, while the red areas are mostly controlled by the opposition. The player is best advised to pass into a teammate positioned within a blue zone.

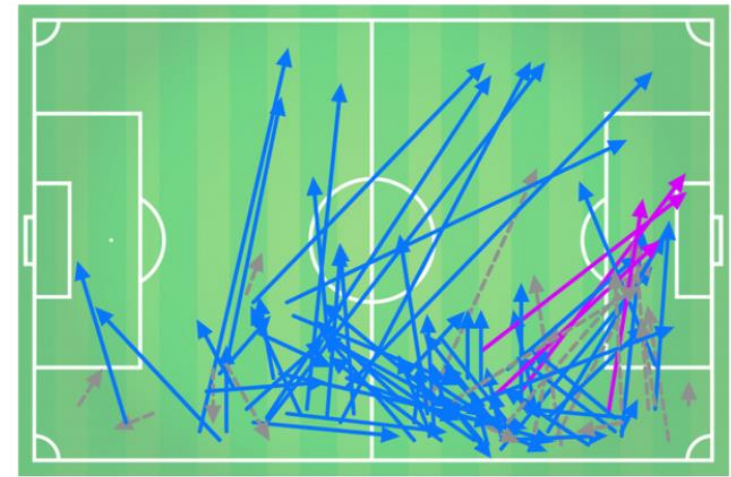
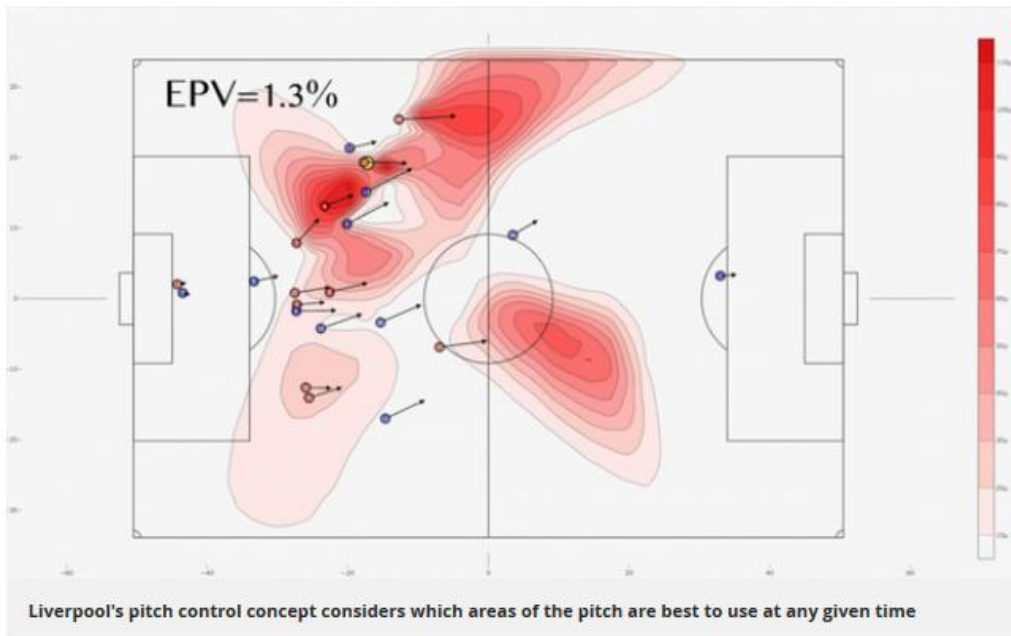
# Ejemplo- Fútbol

$$f(X) = y$$

ib_var_1	ib_var_2	ib_var_3	icn_var_22	icn_var_23	icn_var_24	ico_var_25	ico_var_26	ico_var_27	if_var_65	if_var_68	if_var_80	ob_target
0	1	0	2	1	2	4	4	4	31	4626	0.7666667	0
0	1	0	3	1	1	5	5	4	34	822.4	0.7	0
0	0	0	4	1	1	4	4	4	50	1028	0.6666667	0
0	1	0	2	1	3	4	4	4	30	4112	0.7666667	0
0	0	0	3	1	1	4	4	4	37	1028	0.7333333	0
0	1	0	2	1	1	5	5	5	34	1543.5	0.6	0
0	1	0	3	1	3	5	5	3	27	5140	0.8666667	0
0	0	0	4	1	1	5	4	4	27	1542	0.7333333	0
0	1	0	2	5	3	4	3	5	39	514	0.5333333	0
0	0	0	3	1	1	4	4	4	39	1542	0.6	0
0	1	0	3	1	2	4	5	4	39	6168	0.8	0
0	0	0	1	1	1	4	4	4	40	1028	0.6666667	0
0	1	0	2	1	1	4	5	3	37	2056	0.7666667	0
0	1	0	2	1	1	4	4	4	30	6168	0.7	1
0	0	0	2	1	1	5	5	5	52	514	0.8	0
1	0	0	2	1	1	4	4	4	48	205.6	0.8	1
0	0	0	3	1	2	5	5	5	38	1028	0.6	1
0	1	0	2	1	3	4	4	3	26	805.6	0.7	0
0	0	0	2	1	1	4	4	4	57	1542	0.7666667	0
0	1	0	2	1	2	4	3	4	43	3598	0.8	0
0	1	0	2	1	1	4	4	4	39	1028	0.6	0
0	1	0	2	1	1	5	5	5	27	2056	0.6	0
1	1	0	2	1	2	5	5	5	40	1028	0.8	0
0	1	0	2	1	1	4	4	4	42	925.2	0.7333333	0
0	0	0	3	3	1	3	4	5	55	3084	0.6	1
0	1	0	2	1	1	4	5	4	45	616.8	0.7666667	0



# Liverpool are using incredible data science during matches, and effects are extraordinary

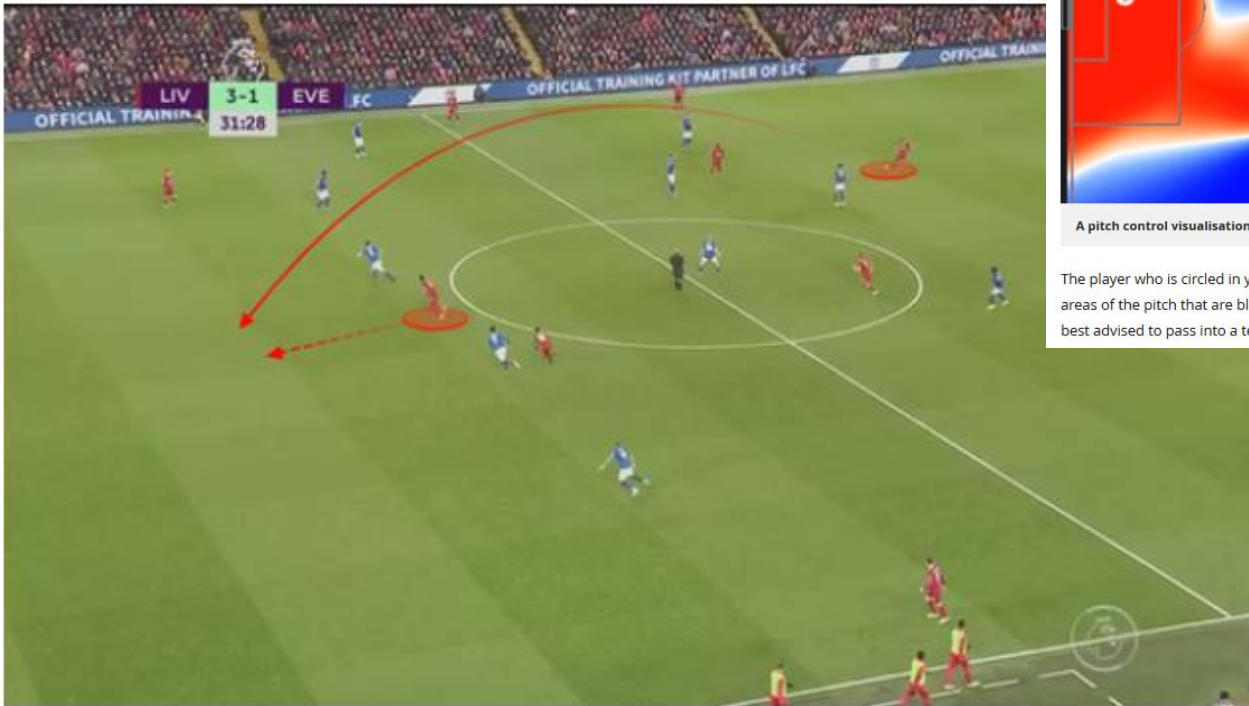


— Accurate — Key pass - - Inaccurate

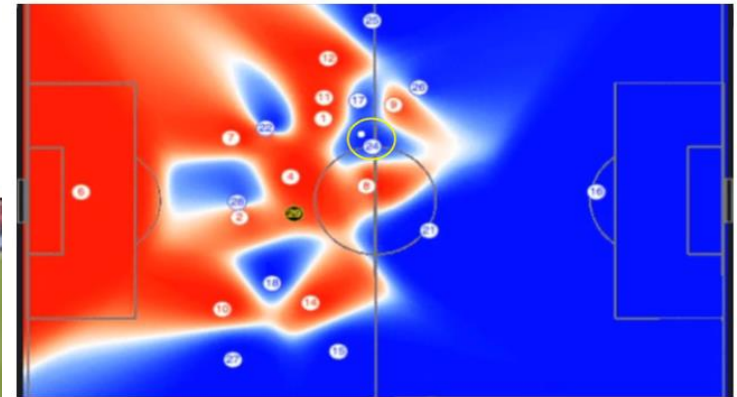
$$f(X) = y$$



# Liverpool are using incredible data science during matches, and effects are extraordinary



Liverpool score against Everton by hitting a simple long pass over the defence



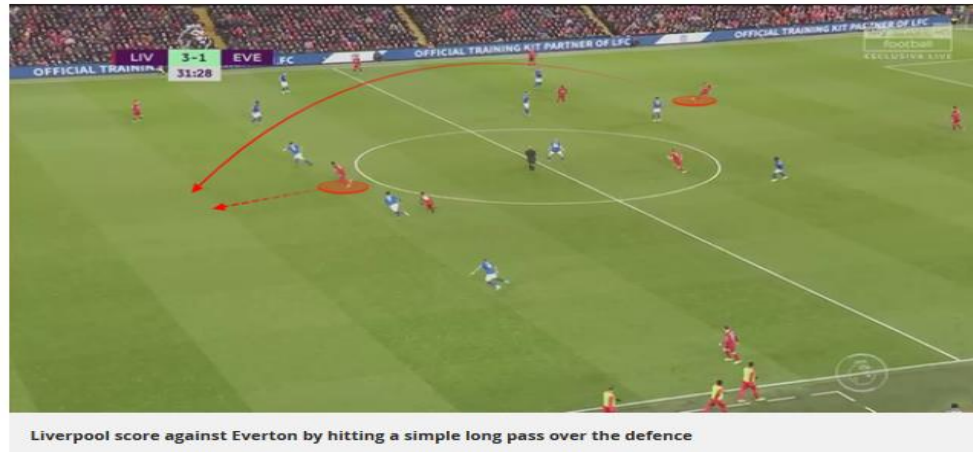
A pitch control visualisation which captures the regions of space controlled by certain players

The player who is circled in yellow has possession of the ball and, essentially, his team has access to the areas of the pitch that are blue, while the red areas are mostly controlled by the opposition. The player is best advised to pass into a teammate positioned within a blue zone.



Jurgen  
Klopp

# Liverpool are Using Incredible Data Science Actual footage:



<https://www.youtube.com/watch?v=rYJD4MMfv1M&t=197s>

$$f(X) = y$$

# JUEGO DE ROLES – Vuelta 3

Adapte el siguiente mensaje para usar chatGPT (por ejemplo en <https://www.bing.com/> o [www.you.com](http://www.you.com)) para generar una base de datos y un modelo de aprendizaje automático para su ejemplo, el que publicó propuesto en el 5.2; ¡Las columnas y la variable de destino deben adaptarse a su ejemplo!

Ejecute el código Python generado por chatGPT en el cuaderno Jupyter de Anaconda o en Google Colab; luego haga una publicación repospodando en este hilo con el código Python generado por chatGPT, agregue a su publicación tanto el conjunto de datos (llamado fintech.xlsx) como el modelo (archivo llamado model.joblib) que se crean después de ejecutar el código.

```
---
Ejemplo del Prof. Manoel Gadi - :
País: España Empresa: Banco Santander
Tarea: Actividades de recolección y
recuperación
Aprendizaje automático: supervisado -
clasificación
La población debería estar formada por:
Clientes existentes en España con deudas
impagadas
Características:
tiempo como cliente
Total pagado en los últimos 12 meses
Deuda total en mora
La variable objetivo debe estar formada por:
1 - bueno - cliente que pagó más del 90% de la
deuda en mora dentro de los 6 meses siguientes
contados a partir de la fecha de vencimiento
del pago.
0 - malo - todos los demás.
```

---- INICIO PROMPT ---

Cree un código Python para una regresión logística utilizando un marco de datos de tres columnas.

Primera columna Tiempo como cliente como variable discreta, segunda columna Total Pagado en los últimos 12 meses como variable continua y tercera columna Deuda Total en Mora como variable continua.

Objetivo binario variable de pagar o no pagar la deuda.

Genere todas las variables aleatoriamente usando numpy.

Establezca la semilla para que todas las variables obtengan siempre el mismo valor.

Guarde los datos en un Excel llamado fintech.xlsx sin índice.

Divida los datos en tren y prueba.

Valide el modelo utilizando el informe de clasificación tanto en los datos del tren como en los de prueba.

Vuelque el modelo en un archivo usando joblib, llame al archivo model.joblib

---- FINAL PROMPT ---

DESCANSO

NOS VEMOS EN 20 MINUTOS!

# Model Deployment in Production

With Python Flask at

[www.pythonanywhere.com/](http://www.pythonanywhere.com/)

<http://tobesure.pythonanywhere.com/>

## Buscador de empresas

Entre el nombre o el CIF de la empresa...

☐ Check for JSON (default HTML)

☐ Check for modelo (default sin modelo)

Buscar

# invoking\_api\_tobesure.py

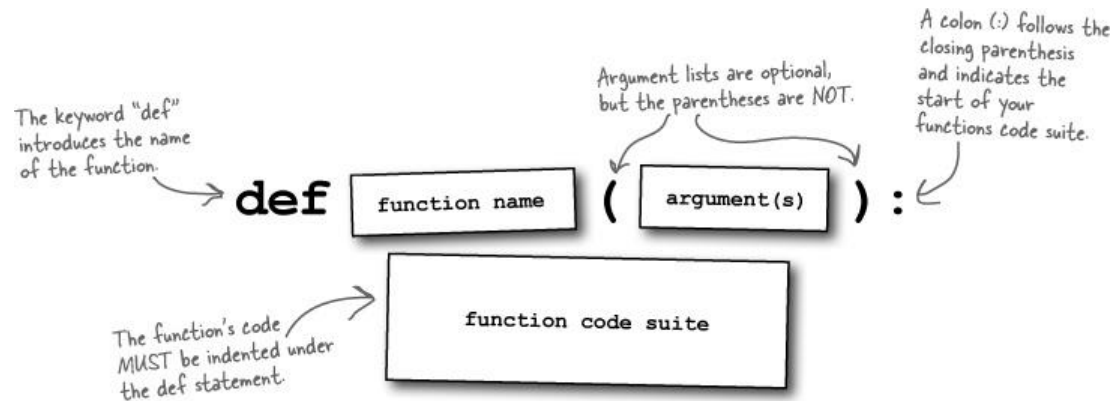
```
from urllib.request import urlopen
sourceCode =
    urlopen("http://tobesure.pythonanywhere.com/buscador
?nif=A28015865&JSON=yes&modelo=yes").read()

import json
empresas = json.loads(sourceCode)

print(empresas[0]['Nombre'])
print(empresas[0]['probabilidad_default'])
```

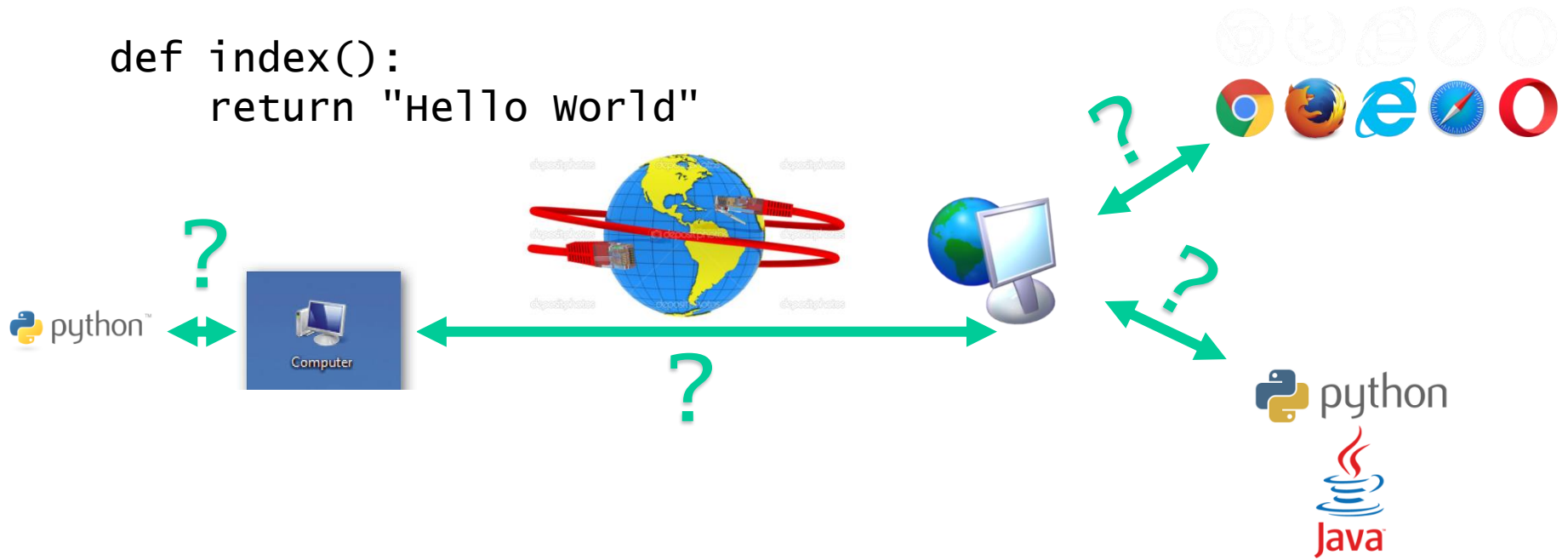


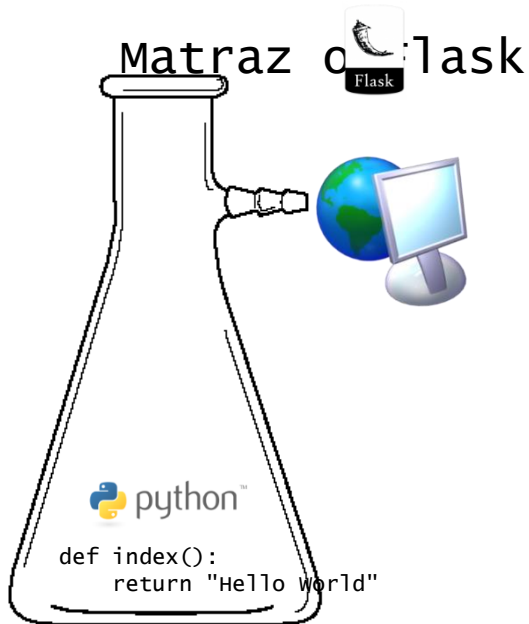
```
def index():  
    return "Hello world"
```





```
def index():  
    return "Hello world"
```





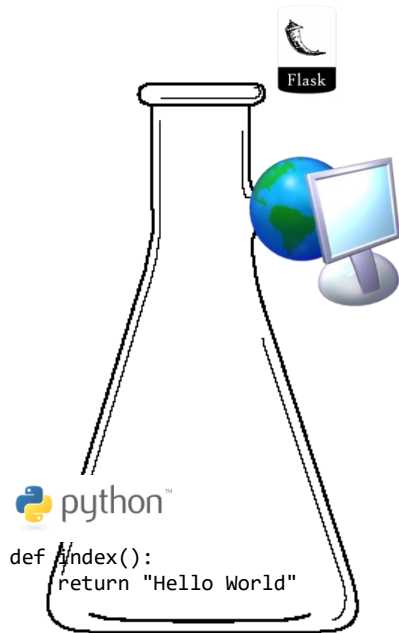
```
# Using both methods of registering  
URLs here  
# just to show that both work
```

```
def handle_route2():  
    return X().route2()
```

```
app.add_url_rule('/y/',  
view_func=handle_route2)
```

```
# the most common is using a decorator:  
@app.route('/x/')  
def handle_route1():  
    return X().route1()
```

## Matraz o Flask



```
from flask import Flask
```

```
app = Flask(__name__)
```

```
@app.route('/')  
def index():  
    return "Hello World"
```

```
@app.route('/miruta1')  
def miruta1():  
    return "Mi ruta 1"
```

```
@app.route('/mivar/<mivar1>')  
def mivar(mivar1):  
    return "mivar="+ mivar1
```

```
if __name__ == '__main__':  
    app.run(  
        port=int("5000")  
    )
```

```

from flask import Flask

app = Flask(__name__)

@app.route('/')
def index():
    return "Hello World"

@app.route('/miruta1')
def miruta1():
    return "Mi ruta 1"

@app.route('/mivar/<mivar1>')
def mivar(mivar1):
    return "mivar="+ mivar1

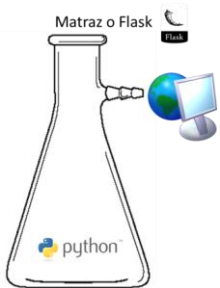
if __name__ == '__main__':
    app.run(
        port=int("5000")
    )

```

**http**://localhost:5000/mivar/mystring1234

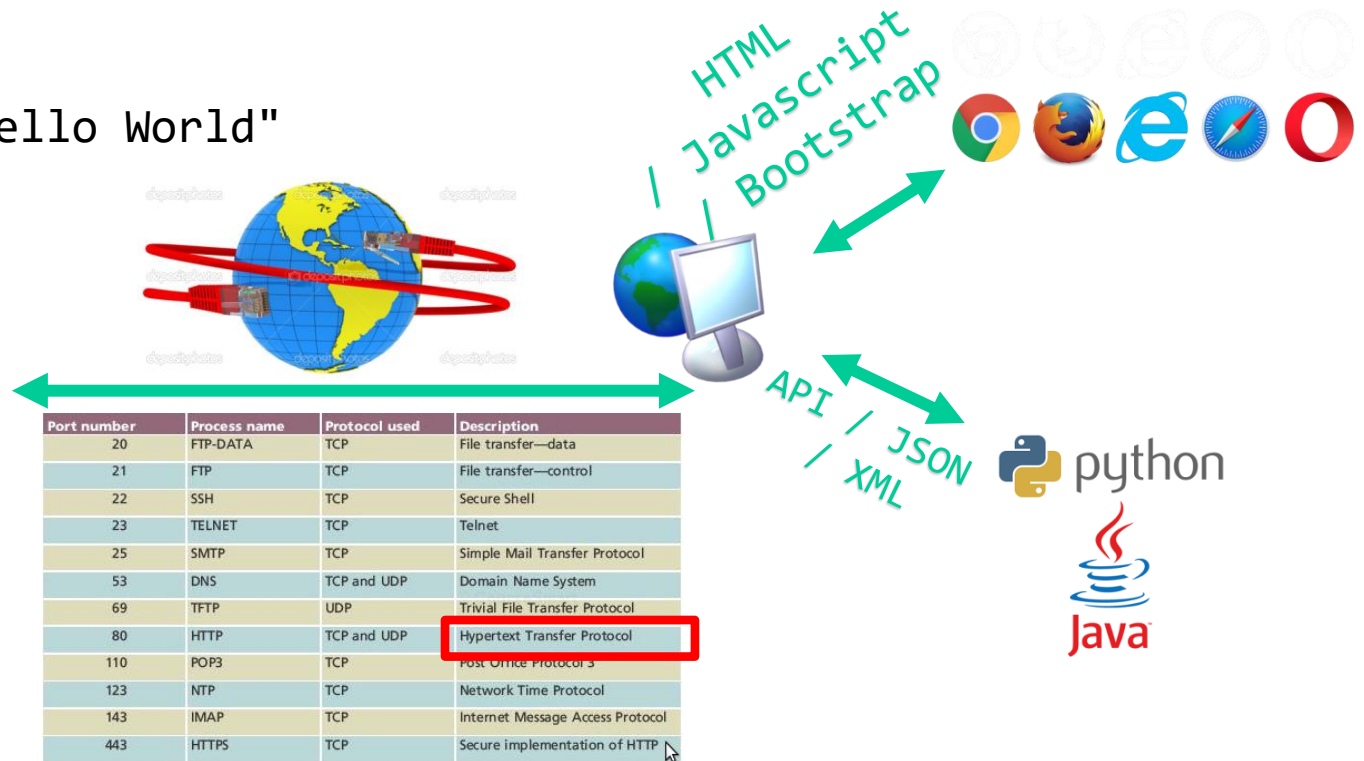
- **http** - http means your data will be transfered using Hypertext Transfer Protocol, most commonly used over the internet
- **5000** - it means you will use the Flask default port (80 is the default internet port)
- **localhost** resolves to 127.0.0.1
- **/mivar** will call the python function “def mivar(mivar1)” with “mystring1234” as value, equivalent to in Python:  

```
print mivar("mystring1234")
```

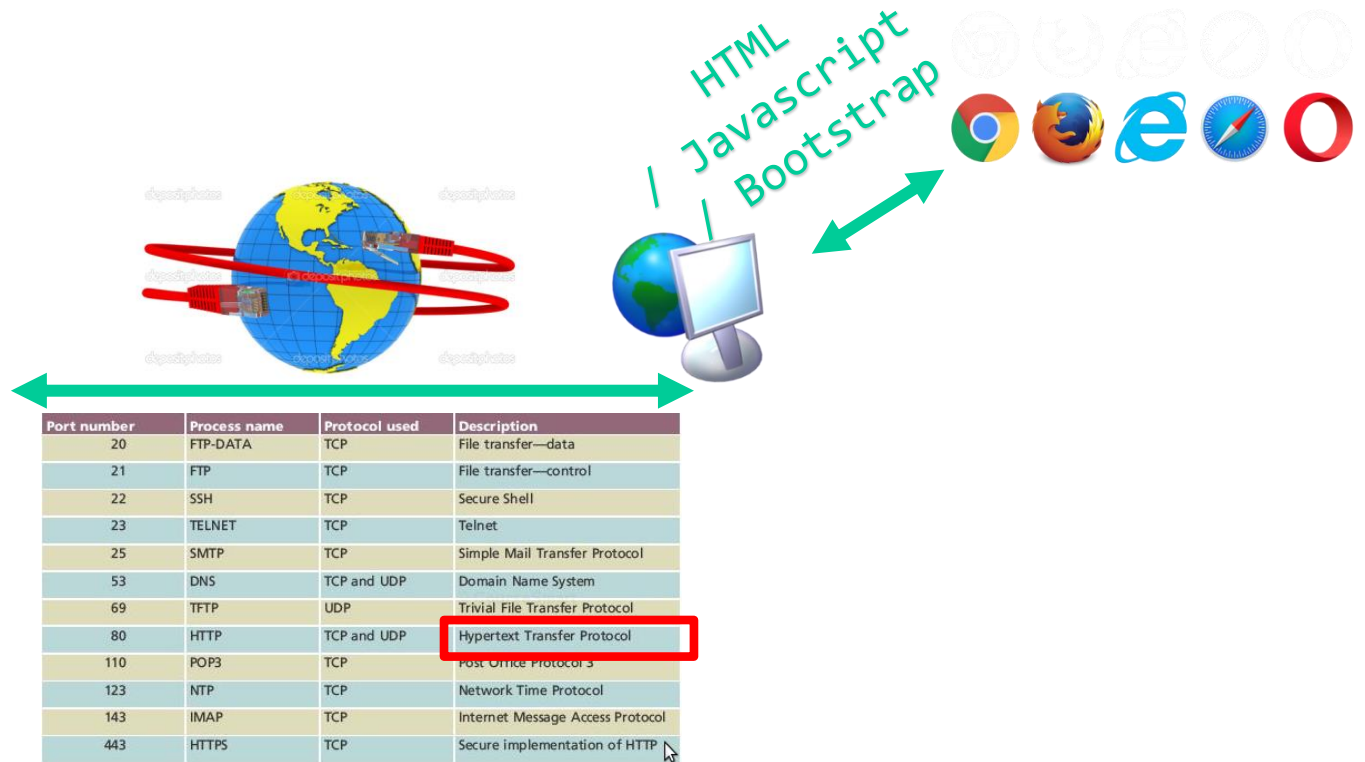


```
def index():
    return "Hello World"
```

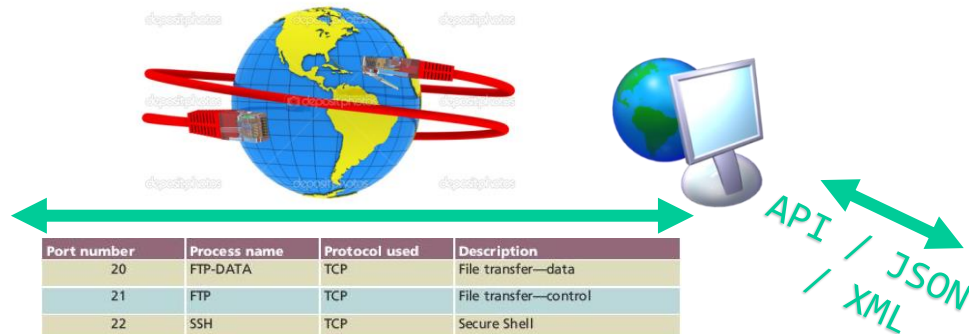
 pythonanywhere



pythonanywhere

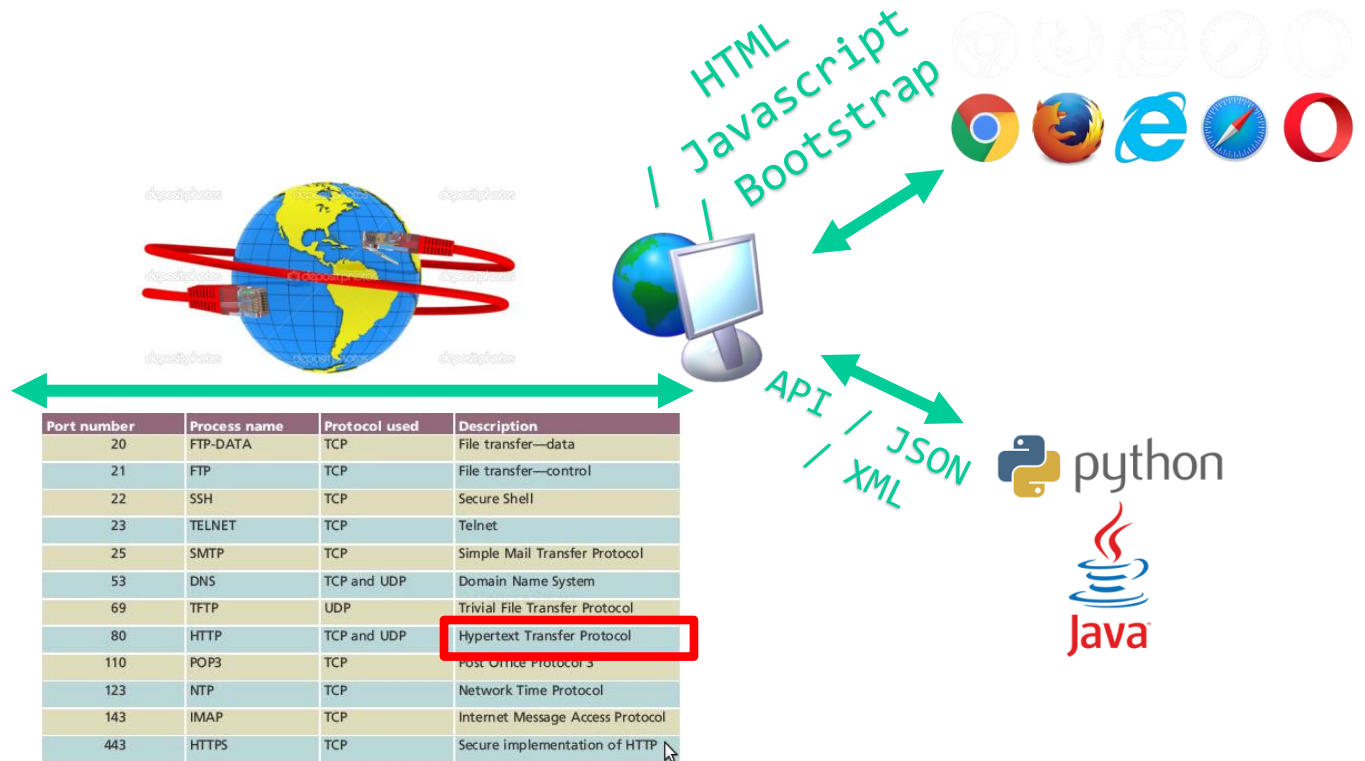


pythonanywhere

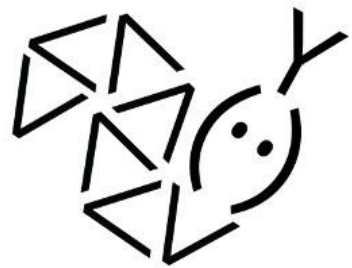


Port number	Process name	Protocol used	Description
20	FTP-DATA	TCP	File transfer—data
21	FTP	TCP	File transfer—control
22	SSH	TCP	Secure Shell
23	TELNET	TCP	Telnet
25	SMTP	TCP	Simple Mail Transfer Protocol
53	DNS	TCP and UDP	Domain Name System
69	TFTP	UDP	Trivial File Transfer Protocol
80	HTTP	TCP and UDP	Hypertext Transfer Protocol
110	POP3	TCP	Post Office Protocol 3
123	NTP	TCP	Network Time Protocol
143	IMAP	TCP	Internet Message Access Protocol
443	HTTPS	TCP	Secure implementation of HTTP

pythonanywhere







pythonanywhere

**Thanks.**

