Informing business decisions with machine learning: a case study

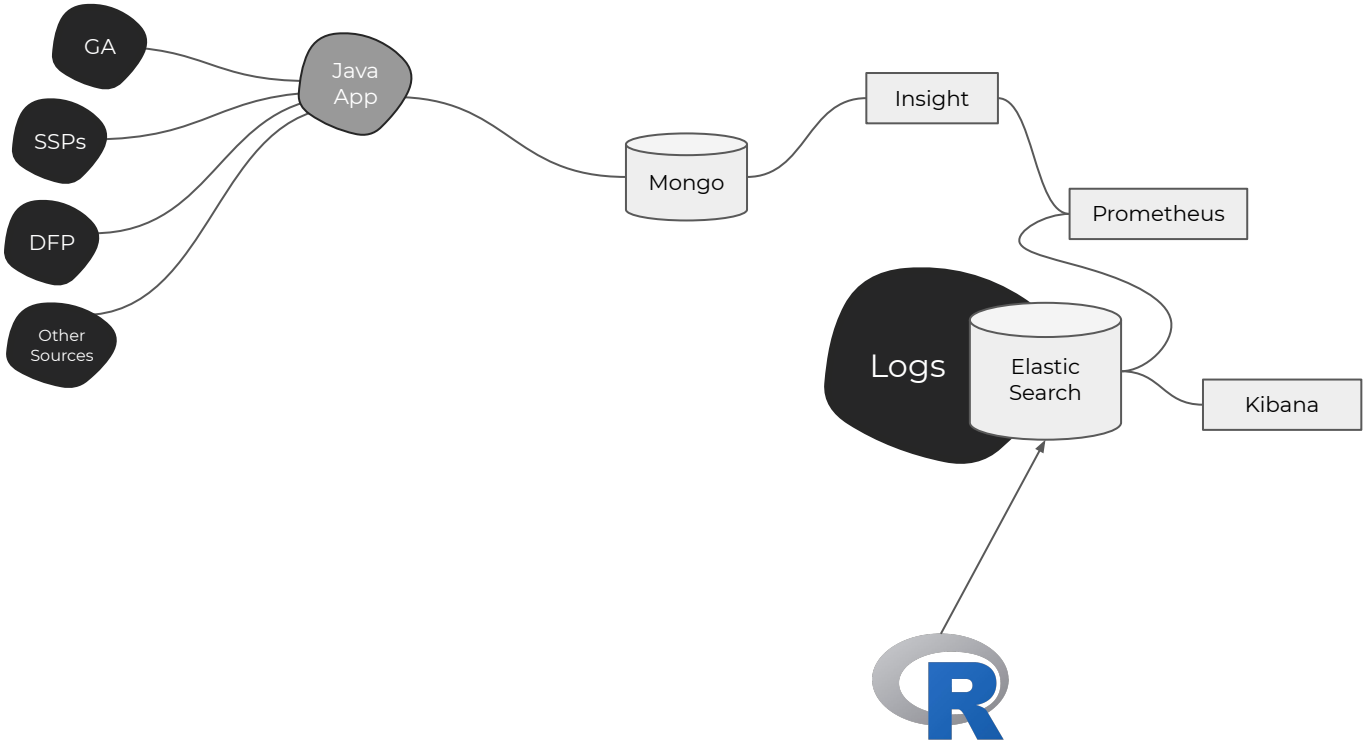
Pol Blasco

Marfeel & Data

Marfeel

- What is it?
    - A free publisher platform
    - Publisher articles are crawled, and rendered with an fast-optimized UX with tones of tech behind, and new monetization technology
    - Marfeel manages the monetization, and shares the revenue with the publisher
- What publishers get out of it?
    - Reduced IT costs, fast mobile web, top-notch web and ad tech, and $$$
- Numbers
    - Works with 600 publishers
    - Manges 1B monthly visits
    - 3TB / day

Marfeel Data (6m ago)



A group by and three counts

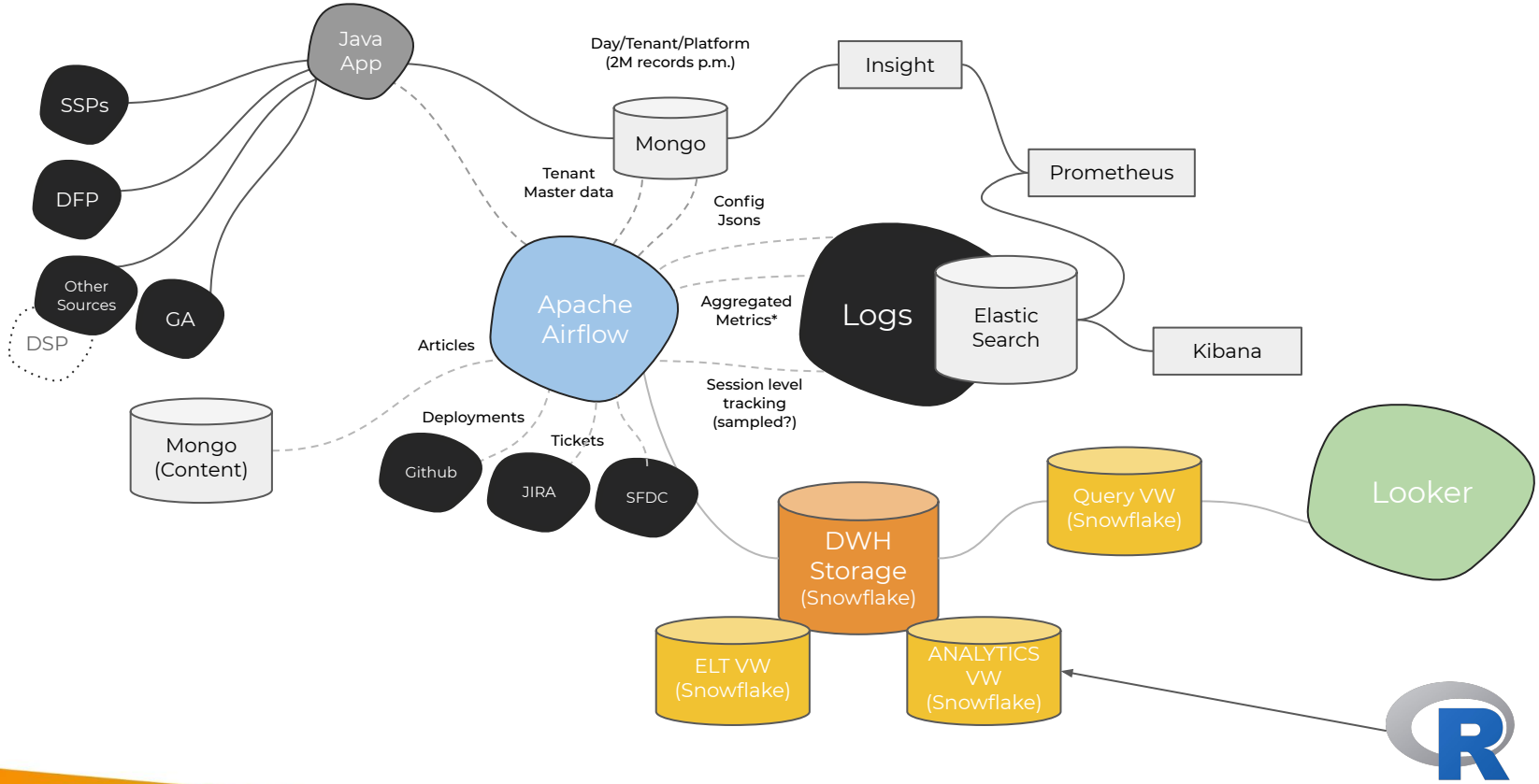
```
{
  "aggs": {
    "2": {
      "date_histogram": {
        "field": "ts",
        "interval": "1d",
        "time_zone": "Europe/Berlin",
        "min_doc_count": 1
      },
      "aggs": {
        "3": {
          "filters": {
            "filters": {
              "sessions with swipe": {
                "query_string": {
                  "query": "interaction_action: swipe",
                  "analyze_wildcard": true,
                  "default_field": "*"
                }
              },
              "sessions with successful swipe": {
                "query_string": {
                  "query": "interaction_action: swipe &&
interaction_action_detail: successful",
                  "analyze_wildcard": true,
                  "default_field": "*"
                }
              },
              "Other Events": {
                "query_string": {
                  "query": "!interaction_action: swipe",
                  "analyze_wildcard": true,
                  "default_field": "*"
                }
              }
            }
          },
          "aggs": {
            "1": {
              "field": "sid"
            }
          }
        }
      }
    }
  },
  "size": 0,
  "_source": {
    "excludes": []
  },
  "stored_fields": [
    "*"
  ],
  "script_fields": {},
  "docvalue_fields": [
    {
      "field": "ts",
      "format": "date_time"
    }
  ],
  "query": {
    "bool": {
      "must": [
        {
          "match_phrase": {
            "type": {
              "query": "user_event"
            }
          }
        },
        {
          "match_phrase": {
            "mds": {
              "query": "marfeel_browser"
            }
          }
        },
        {
          "match_phrase": {
            "dh": {
              "query": "m.washingtontimes.com"
            }
          }
        },
        {
          "range": {
            "ts": {
              "gte": 1580855126660,
              "lte": 1580856026660,
              "format": "epoch_millis"
            }
          }
        },
        {
          "match_phrase": {
            "type": {
              "query": "user_event"
            }
          }
        },
        {
          "match_phrase": {
            "mds": {
              "query": "marfeel_browser"
            }
          }
        },
        {
          "match_phrase": {
            "dh": {
              "query": "m.washingtontimes.com"
            }
          }
        }
      ],
```

ElasticSearch + R

- Kibana is great
- No powerful aggregation framework, no joints, no filtering by large arrays.
- Data retrieval is slow process
- Elastic package
  - Provides API connection
  - Pagination, error handling, authentification...
  - Does NOT help writing the queries
- MrfElastic
  - Helps building some ES queries
  - Still very faulty

```
baseQuery <- TERM_filter("mdt", "s") %>%
 append(TERM_filter("mds", "marfeel_browser")) %>%
 append(RANGE_filter("ts", fromTs, toTs)) %>%
 append(OR_operator("dh", tenantList))
```

Marfeel Data (today)



SnowFlake + R

- **As fast as you want to pay**
- **Snowflake + R**
  - SQL based DB
  - Native connection with R and dplyr (here, here, and here)

SnowFlake + R

```
library("RJDBC")
library("dplyr")
library("dplyr.snowflakedb")

my_db <- src_snowflakedb(user = "POL_BLASCO", password = XXXXXX)
UJ_interactions <- tbl(my_db, "UJ_RETENTION_NAVIAGTION_BY_CID")

#UJ_interactions %>% mutate() %>% goup_by
#This is data frame which you can use dplyr and it will translate it to SQL and execute in snowflake.
```

- **collect(): download the table from snowflake to where R code is executed**
- **compute(): store the result of a query in snowflake**
- **collapse(): get the SQL query that will execute**

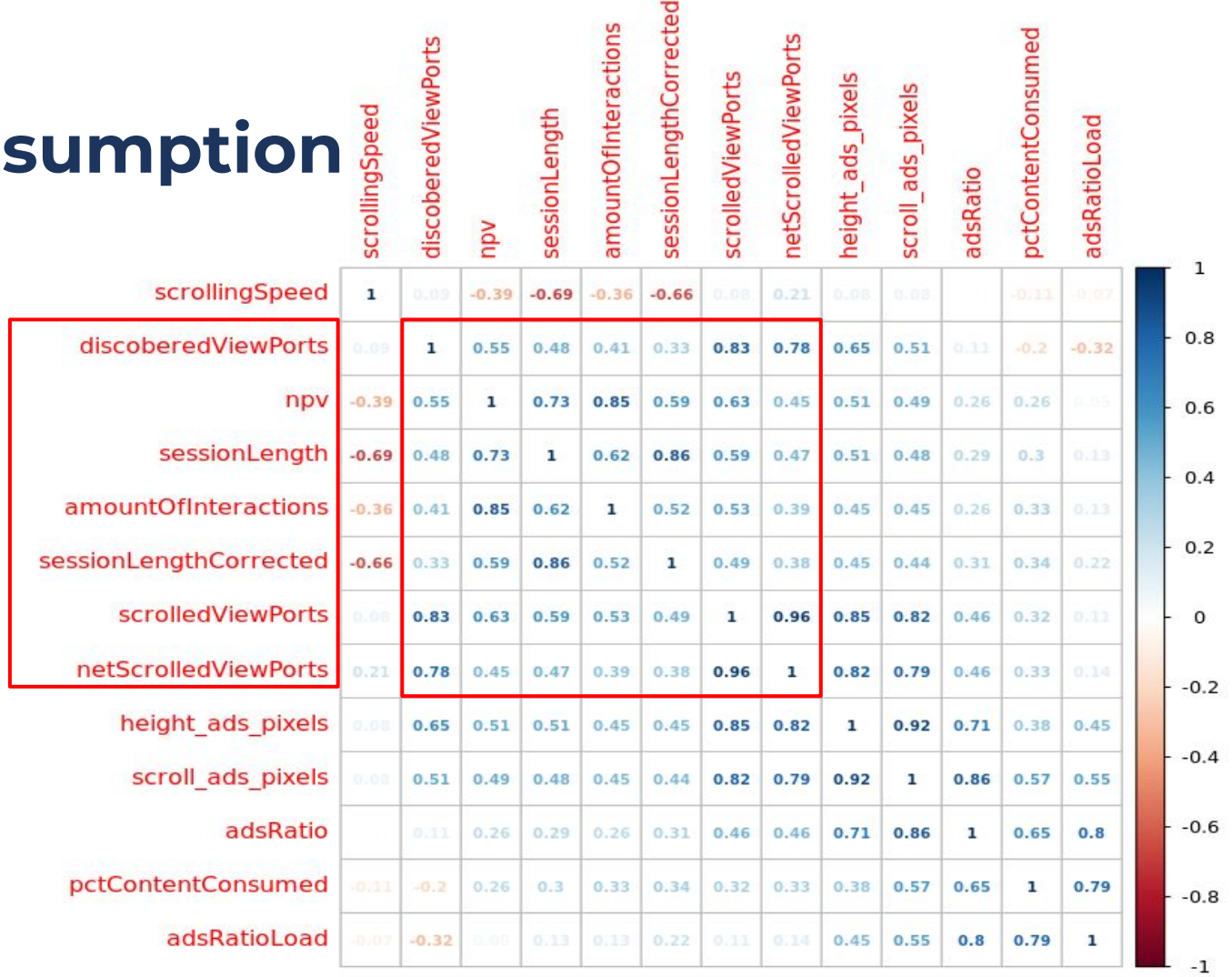
The Question

I want to know how to measure the success of UX changes....

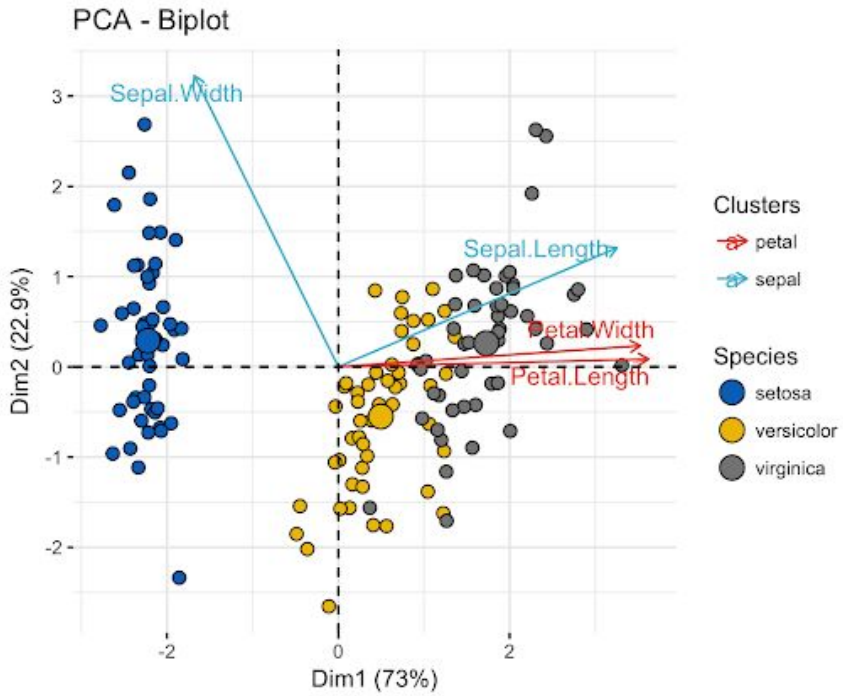
Successful UX changes

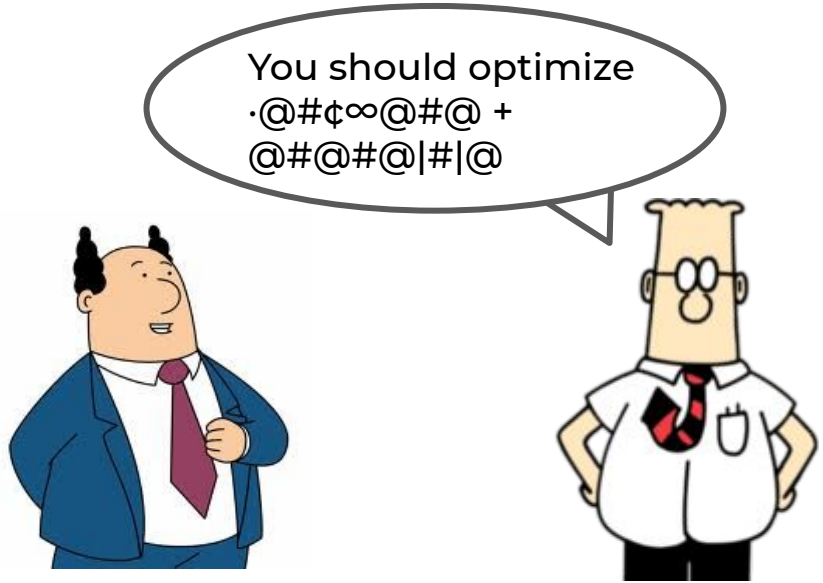
- Readers are more engaging
- Content consumption metrics go up
  - More page views
  - More reading time
  - More scroll down
  - More “high value” actions
  - ....

Content Consumption

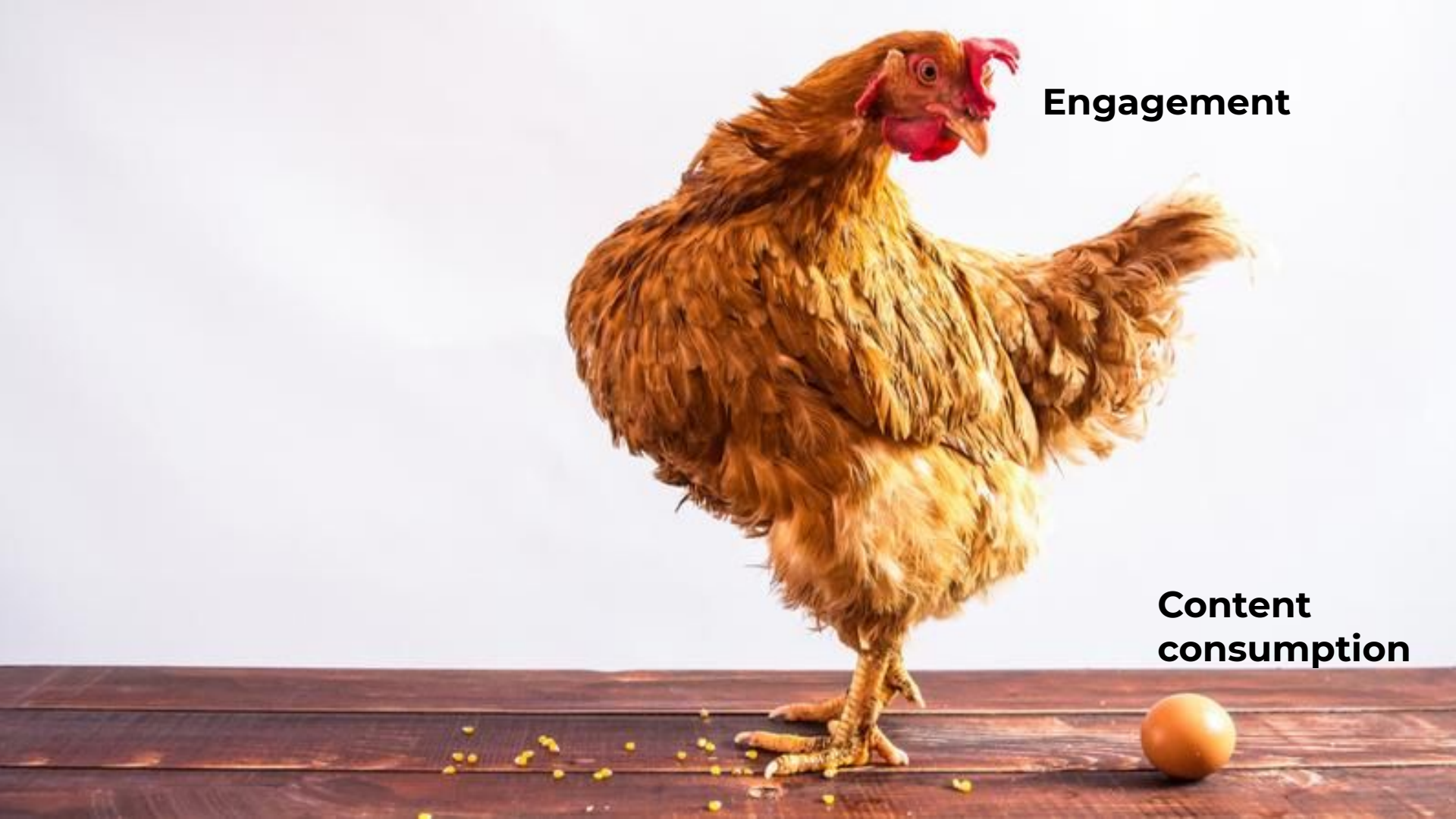


PCA


PCA - Biplot
Sepal.Width
Sepal.Length
Petal.Width
Petal.Length
Dim2 (22.9%)
Dim1 (73%)
3
2
1
0
-1
-2
-2
0
2
Clusters
petal
sepal
Species
setosa
versicolor
virginica


You should optimize
·@#¢∞@#@ +
@#@#@|#|@

OPTIMIZE ALL


Engagement
Content consumption

What is engagement?

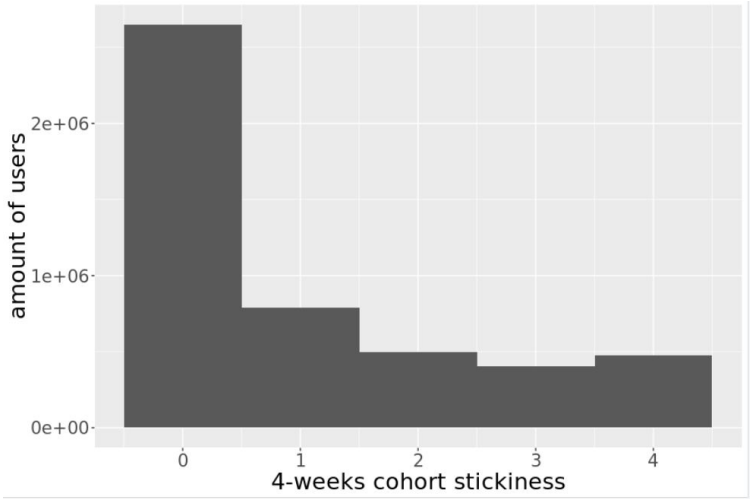
Engaged users come more often

=

Engagement is similar to retention, recurrency, loyalty...

4-weeks cohort stickiness

- *Qualitative*
  - *It does check for a consistent recurrency over time*
  - *It can easily inform about the intensity of the recurrency*
- *Quantitative:*
  - *Shows nice properties such as shape, variance, correlation, etc...*



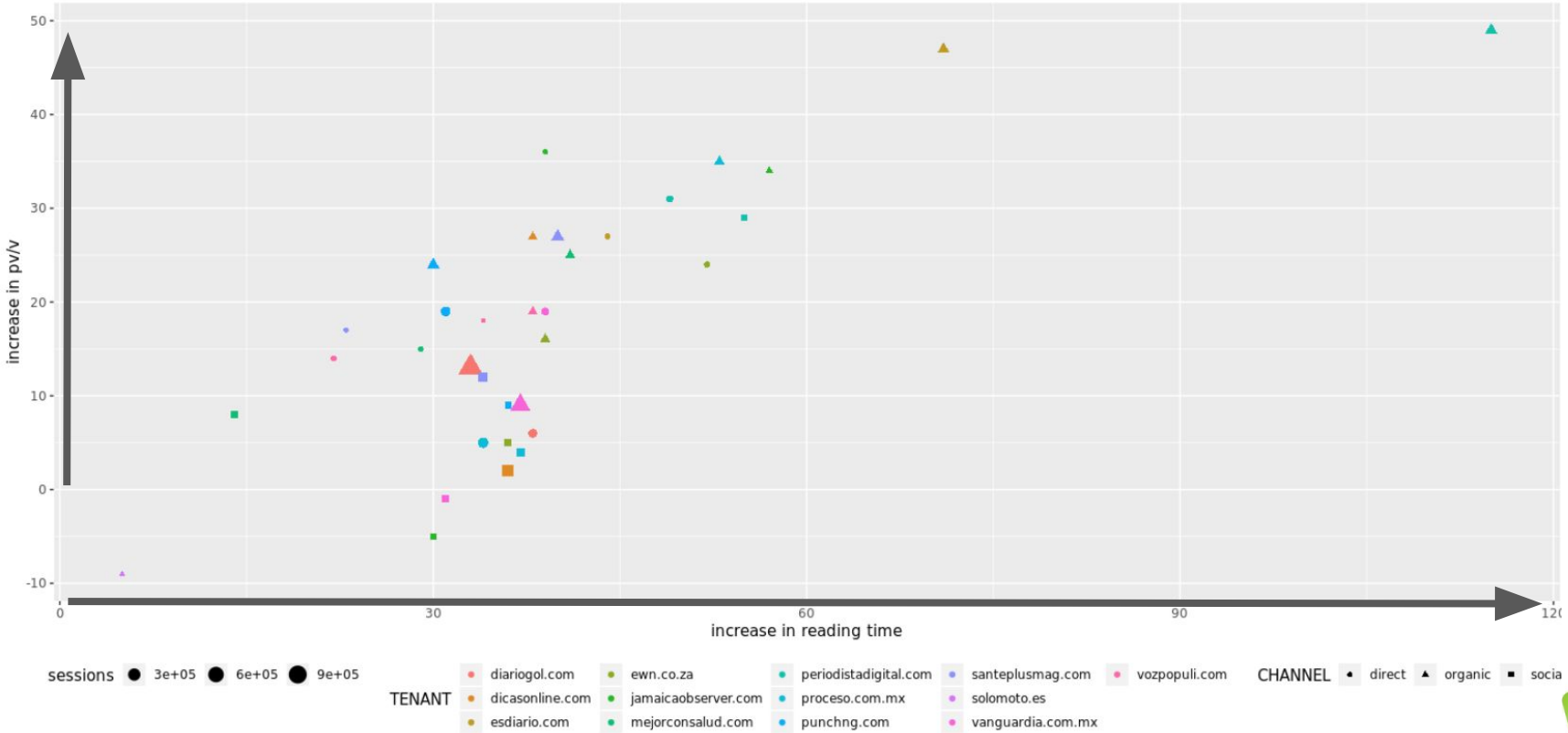
Engaged users navigate different

Recurrent vs non recurrent user

- ***Recurrent users consume more content and more often***
  - *Reading time is 35% larger in recurrent users*
  - *Pageviews / session is 11% larger in recurrent users*

- ***This is consistent across many segments and publishers***

Recurrent users segment



Reframing the Question

*I want to know how to measure the **success** of UX changes....*

1. **Success means that users navigate better**
2. **Engaged / recurrent users navigate better**
3. **Which content consumption feature defines better engagement / recurrence?**

I want to know how to measure the ***which content consumption metric is the best predictor of engagement***

Modeling

Framing the problem into Machine Learning

- For each session we would predict if that session belongs to a user that is engaged or not engaged
- The feature that provides higher predicting power will be the measure of success
- This boils down to a **variable importance analysis of a binary classification problem**

Binary Classification Problem

- Target variable is binary 4-week cohort stickiness:
  - {1} if sessions in 2 or more different weeks
  - {0} if less than 2 sessions in different weeks
- Features
  - **Base features:** Publisher name, acquisition channel of the user, landing page, country, os, os version, network....
  - **Content consumption features:** pv, reading time, scroll, and many more...
- Data
  - 275M sessions over 23 publishers

Content Consumption Features

MONTHLY_SESSIONS	11548227
MONTHLY_USERS	3104381
MONTHLY_RVR	0.66
MONTHLY_RSR	0.68
MONTHLY_SPU	3.72
WEEKLY_SESSIONS	2668609
WEEKLY_USERS	971738
WEEKLY_RVR	0.65
WEEKLY_RSR	0.5
WEEKLY_SPU	2.75
RETENTION_COHORT_USERS	144365
D7_RETAINED_USERS	71185
D28_RETAINED_USERS	94014
D7_RETENTION	0.49
D28_RETENTION	0.65
D7_EXACT_RETENTION	0.192
D28_EXACT_RETENTION	0.104
NAVIGATION_SESSIONS	767810
NAVIGATION_USERS	616492
TOTAL_PV	2274410
TOTAL_READING_TIME	89406465.15
TOTAL_EPV_3	1367356
TOTAL_EPV_5	1254345
TOTAL_EPV_10	1054351

TOTAL_ENGAGED_READING_TIME_3	88909543.6
TOTAL_ENGAGED_READING_TIME_5	88463993
TOTAL_ENGAGED_READING_TIME_10	86996273.25
AVG_CONTENT_LENGTH_PX	6542.998573
MEDIAN_CONTENT_LENGTH_PX	6354
AVG_CONTENT_SCROLL_PX	1190.240884
MEDIAN_CONTENT_SCROLL_PX	902
AVG_CONTENT_CONSUMPTION	0.18
MEDIAN_CONTENT_CONSUMPTION	0.14
PV_PER_SESSION	2.96
READING_TIME_PER_SESSION	116.44
EPV_3_PER_SESSION	1.78
EPV_5_PER_SESSION	1.63
EPV_10_PER_SESSION	1.37
ENGAGED_READING_TIME_3_PER_SESSION	115.8
ENGAGED_READING_TIME_5_PER_SESSION	115.22
ENGAGED_READING_TIME_10_PER_SESSION	113.3
CONTENT_LENGTH_PX_PER_SESSION	0.01
CONTENT_SCROLL_PX_PER_SESSION	0
PV_PER_USER	3.69
READING_TIME_PER_USER	145.02
EPV_3_PER_USER	2.22
EPV_5_PER_USER	2.03
EPV_10_PER_USER	1.71

ENGAGED_READING_TIME_3_PER_USER	144.22
ENGAGED_READING_TIME_5_PER_USER	143.5
ENGAGED_READING_TIME_10_PER_USER	141.12
CONTENT_LENGTH_PX_PER_USER	0.01
CONTENT_SCROLL_PX_PER_USER	0
NORMALIZED_PV_PER_SESSION	3.645031542
NORMALIZED_READING_TIME_PER_SESSION	143.3882684
NORMALIZED_EPV_3_PER_SESSION	2.191982208
NORMALIZED_EPV_5_PER_SESSION	2.007228846
NORMALIZED_EPV_10_PER_SESSION	1.687032658
NORMALIZED_ENGAGED_READING_TIME_3_PER_SESSION	142.6001861
NORMALIZED_ENGAGED_READING_TIME_5_PER_SESSION	141.8859085
NORMALIZED_ENGAGED_READING_TIME_10_PER_SESSION	139.5215811
NORMALIZED_CONTENT_SCROLL_PX_PER_SESSION	0
NORMALIZED_PV_PER_USER	4.543981937
NORMALIZED_READING_TIME_PER_USER	178.582696
NORMALIZED_EPV_3_PER_USER	2.733753513
NORMALIZED_EPV_5_PER_USER	2.499850762
NORMALIZED_EPV_10_PER_USER	2.105769344
NORMALIZED_ENGAGED_READING_TIME_3_PER_USER	177.5975328
NORMALIZED_ENGAGED_READING_TIME_5_PER_USER	176.71091
NORMALIZED_ENGAGED_READING_TIME_10_PER_USER	173.7800755
NORMALIZED_CONTENT_SCROLL_PX_PER_USER	0

Variable Importance

- Random forest perform variable importance analysis at (nearly) no cost.
- This method present several issues: 1, 2, and 3
  - Trees are biased towards categorical features with large number of levels
  - Collinear / monotonic features have a un realistic feature importance score
  - One must use the permutation feature importance

Feature Selection

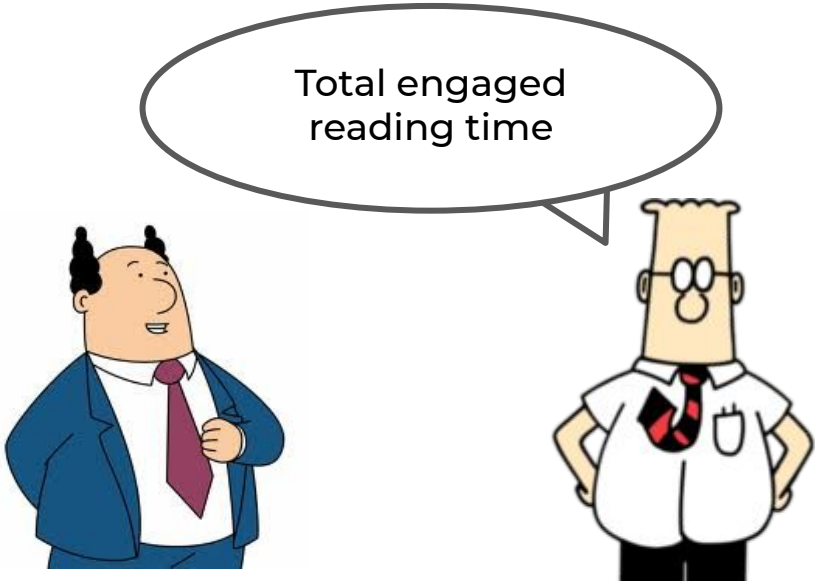
(single) stepwise forward feature selection with repetition

1. **Repeat several times**
    a. Fit and evaluate a classifier that uses only the **base features**
    b. For each feature in **content consumption features list**
        i. Add feature to the model
        ii. Fit the best parameters for the model by CV
        iii. Measure the increase in accuracy
    c. Compare accuracy increase of all features and select the one with larger increase
2. **List the winning ratio of each feature**

TOP-10 features

Content Consumption Metric	Pct Winning Ratio
TOTAL_ENGAGED_READING_TIME_3	66.2%
TOTAL_ENGAGED_READING_TIME_5	63.8%
TOTAL_READING_TIME	56.2%
VIEW_PORTS_SCROLLED	51.9%
PCT_CONTENT_SCROLLED	48.8%
NORMALIZED_READING_TIME	47.7%
TOTAL_EPV_3	47.3%
TOTAL_EPV_5	45.0%
TOTAL_PV	45.0%
TOTAL_EPV_10	38.5%

Magic



Questions?