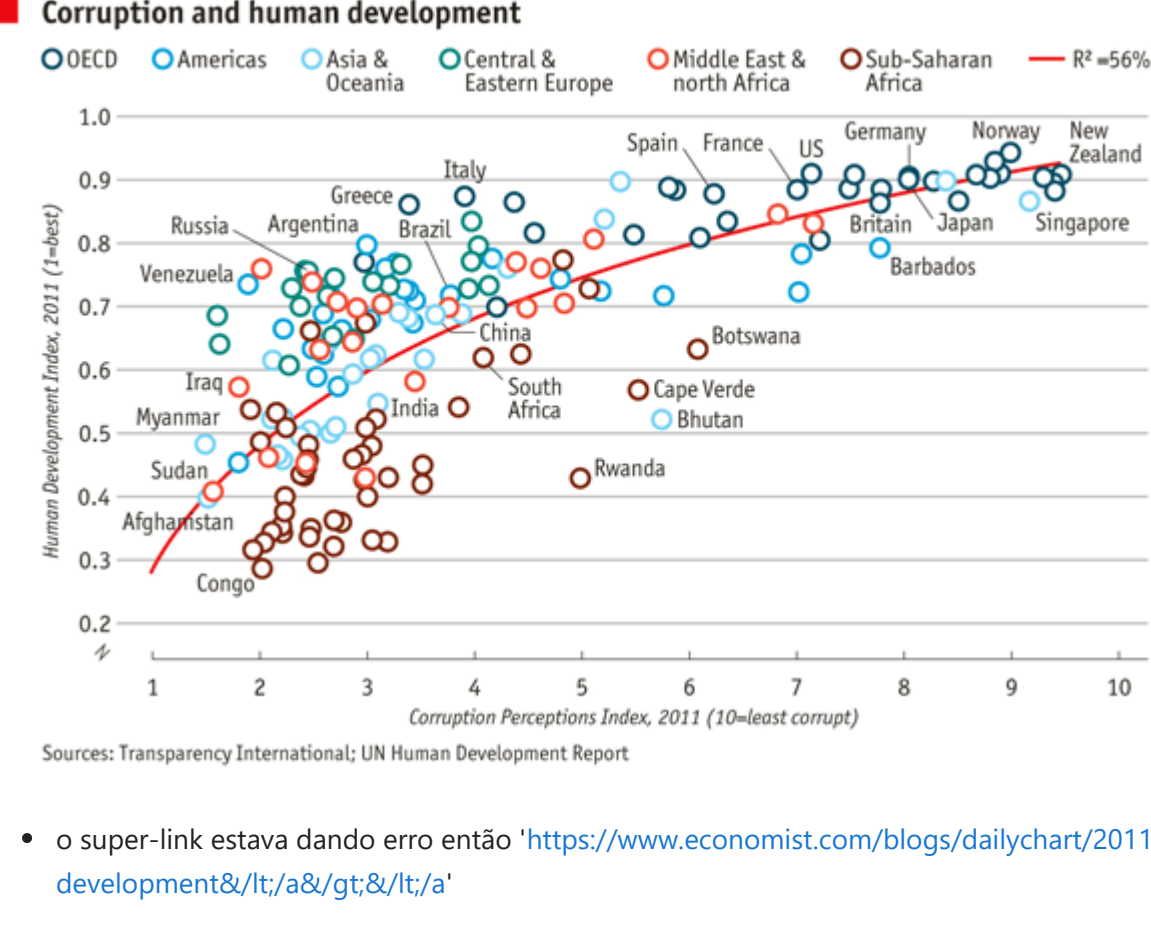


Visualização Escolhida

Eu escolhi a visualização do jornal The Economist que em 2011 fez o seguinte grafico:



- o super-link estava dando erro então <https://www.economist.com/blogs/dailychart/2011/12/corruption-and-development&/t/a&/gt;&/t/a/>

Obtenção dos dados

Eu baixei os dados de:

- IDH
- Gini
- Países por continente

Eu peguei os dados de 2011 e trabalhei com eles.

Os Dados vieram bem complicados, deu um trabalhozinho para configura-los.

O processo completo está no [GitHub](#)

```
In [1]: import numpy as np
import pandas as pd

In [2]: df = pd.read_csv('Dados\dados2011.csv', sep=',')
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 176 entries, 0 to 175
Data columns (total 10 columns):
#   Column      Non-Null Count  Dtype
---  --
0   Unnamed: 0   176 non-null    int64
1   country      176 non-null    object
2   region       176 non-null    object
3   CPI          176 non-null    float64
4   IDH          176 non-null    float64
5   ano         176 non-null    int64
6   Texto       176 non-null    object
7   line        176 non-null    object
8   continent   176 non-null    object
9   sub_region  176 non-null    object
dtypes: float64(2), int64(2), object(6)
memory usage: 13.9+ KB

In [3]: from sklearn.metrics import r2_score
round(100*abs(r2_score(df['IDH'], np.log10(df['CPI']))))

Out[3]: 56
```

Fazer a Visualização

```
In [4]: import altair as alt
```

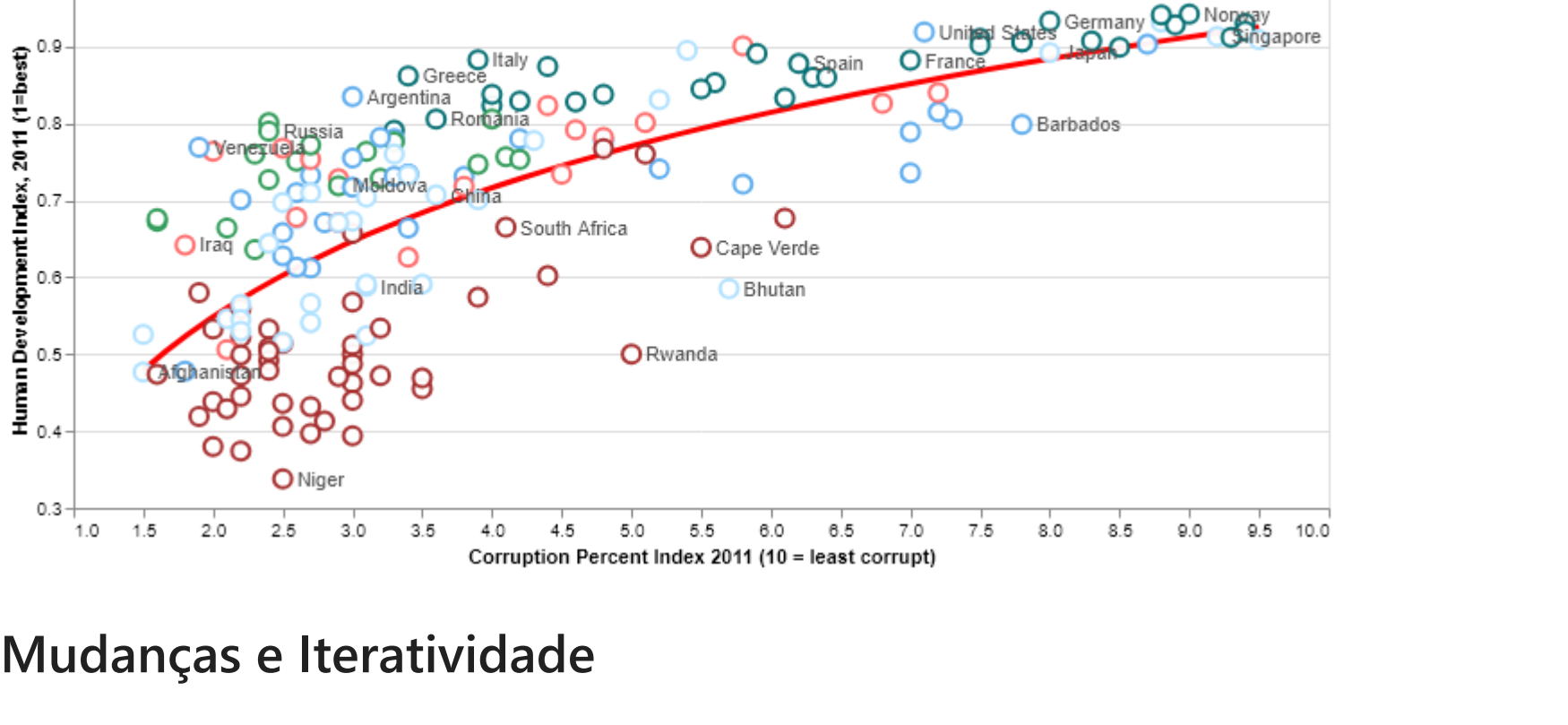
```
In [5]: base = alt.Chart(df, title = 'Corruption and Human Development').encode(
    x = alt.X('CPI:Q', title='Corruption Percent Index 2011 (10 = least corrupt)',
    scale=alt.Scale(zero=False), axis=alt.Axis(grid=False)),
    y = alt.Y('IDH:Q', title = 'Human Development Index, 2011 (1=best)', scale=alt.Scale(zero=False, domain=[.3,
    ]).properties(
    width=700,
    height=300,
    ))

points = base.mark_point(size=100, fill='white', opacity=0.95).encode(color = alt.Color('region',
    scale=alt.Scale(range=[ "#5cacee", "#b0e2ff", "#2e9b57", "#ff6a6a", "#006d71", "#a52a2a", "#ff6347"])),
    legend=alt.Legend(
    orient='none',
    legendY=-20,
    legendX=-20,
    direction='horizontal',
    title=None,
    labelFontSize = 11)))

text = base.mark_text(
    align='left',
    baseline='middle',
    dx=8,
    dy=1,
    color='#555555',
    size = 11
).encode(
    text='Texto'
)

curve = base.mark_line(
    color='red',
    size=3
).transform_regression(
    'CPI',
    'IDH',
    method='log'
).encode(opacity=alt.Opacity('line', legend=alt.Legend(orient='none', legendY=-20, legendX=635, direction='horizontal',
    title=None, values=['R^2 = 56%']), scale=alt.Scale(type='ordinal', range=[1])))

curve + points + text
```



Mudanças e Iteratividade

Mudanças que quero fazer:

- Particularmente eu não gosto muito desse arranjo de cores, vou fazer algumas alterações consultando a ferramenta do AdobeColor
- Colocar uma tooltip que mostra o nome do país, IDH e o Índice de Corrupção.
- Colocar destaque em países de determinado continente quando selecionado na legenda.

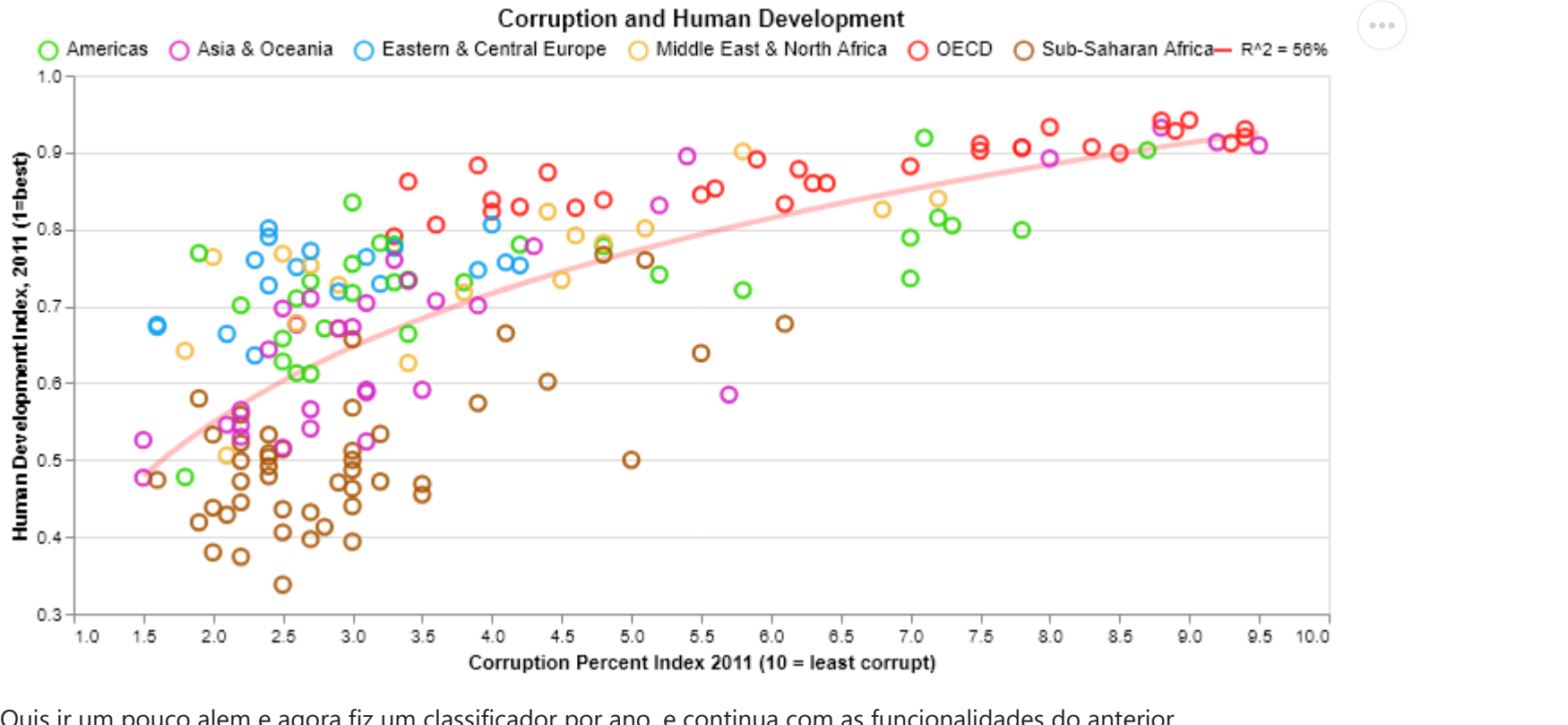
```
In [6]: selection = alt.selection_multi(fields=['region'], bind='legend')
scales = alt.selection_interval(bind='scales')

base_int = alt.Chart(df, title = 'Corruption and Human Development').encode(
    x = alt.X('CPI:Q', title='Corruption Percent Index 2011 (10 = least corrupt)',
    scale=alt.Scale(zero=False), axis=alt.Axis(grid=False)),
    y = alt.Y('IDH:Q', title = 'Human Development Index, 2011 (1=best)', scale=alt.Scale(zero=False, domain=[.3,
    ]).properties(
    width=700,
    height=300,
    ))

points_int = base_int.mark_point(size=70).encode(color = alt.Color('region', scale=alt.Scale(
    range=[ "#2CD602", "#D625C4", "#0F9FEE", "#F5B938", "#FF1C14", "#A85402", "#FF6347" ]
    )), legend=alt.Legend(
    orient='none',
    legendY=-20,
    legendX=-20,
    direction='horizontal',
    title=None,
    labelFontSize = 11),
    opacity=alt.condition(selection, alt.value(.9), alt.value(0.1)),
    tooltip=[ 'country', 'CPI', 'IDH', 'region' ]
    ).add_selection(
    selection,
    scales
    )

curve_int = base_int.mark_line(
    color='red',
    size=3
).transform_regression(
    'CPI',
    'IDH',
    method='log'
).encode(opacity=alt.Opacity('line', legend=alt.Legend(orient='none', legendY=-20, legendX=635, direction='horizontal',
    title=None, values=['R^2 = 56%']), scale=alt.Scale(type='ordinal', range=[0.25])))

curve_int + points_int
```



Quis ir um pouco alem e agora fiz um classificador por ano, e continua com as funcionalidades do anterior.

```
In [7]: tudo = pd.read_csv('Dados\dados.csv', sep=',').rename(columns = {'ano': 'year', 'IDH': 'HDI'})
tudo
```

Unnamed: 0	country	region	CPI	HDI	year	Texto	line	continent	sub_region
0	Afghanistan	Asia & Oceania	1.5	0.477	2011	Afghanistan	line	Asia	Southern Asia
1	Albania	Eastern & Central Europe	3.1	0.764	2011		line	Europe	Southern Europe
2	Algeria	Middle East & North Africa	2.9	0.728	2011		line	Africa	Northern Africa
3	Angola	Sub-Saharan Africa	2.0	0.533	2011		line	Africa	Middle Africa
4	Argentina	Americas	3.0	0.835	2011	Argentina	line	Americas	South America
...
678	Sudan	SSA	1.6	0.510	2019	NaN	line	Africa	Northern Africa
679	Venezuela	AME	1.6	0.711	2019	NaN	line	Americas	South America
680	Yemen	MENA	1.5	0.470	2019	NaN	line	Asia	Western Asia
681	Syria	MENA	1.3	0.567	2019	NaN	line	Asia	Western Asia
682	South Sudan	SSA	1.2	0.433	2019	NaN	line	Africa	Eastern Africa

683 rows x 10 columns

```
In [8]: select_year = alt.selection_single(
    name='Select', fields=['year'], init={'year': 2019},
    bind=alt.binding_range(min=2011, max=2019, step=2)
)

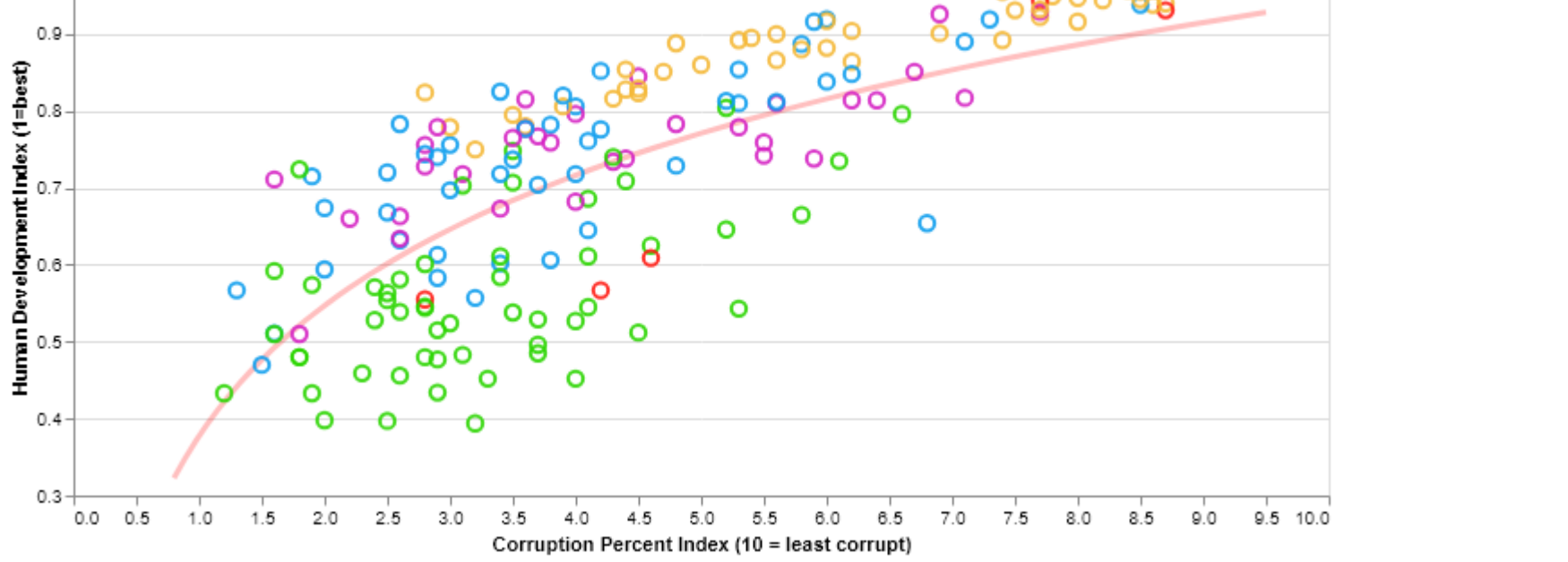
selection = alt.selection_multi(fields=['continent'], bind='legend')
scales = alt.selection_interval(bind='scales')

base_int = alt.Chart(tudo, title = 'Corruption and Human Development').encode(
    x = alt.X('CPI:Q', title='Corruption Percent Index (10 = least corrupt)',
    scale=alt.Scale(zero=False), axis=alt.Axis(grid=False)),
    y = alt.Y('HDI:Q', title = 'Human Development Index (1=best)', scale=alt.Scale(zero=False, domain=[.3, 1])),
    ).properties(
    width=700,
    height=300,
    )

points_int = base_int.mark_point(size=70).encode(color = alt.Color('continent', scale=alt.Scale(
    range=[ "#2CD602", "#D625C4", "#0F9FEE", "#F5B938", "#FF1C14", "#A85402", "#FF6347" ]
    )), legend=alt.Legend(
    orient='none',
    legendY=-20,
    legendX=193,
    direction='horizontal',
    title=None,
    labelFontSize = 11),
    opacity=alt.condition(selection, alt.value(.9), alt.value(0.1)),
    tooltip=[ 'country', 'CPI', 'HDI', 'continent', 'year' ]
    ).add_selection(
    selection,
    scales,
    select_year
    ).transform_filter(select_year)

curve_int = base_int.mark_line(
    color='red',
    size=3,
    opacity=0.25
).transform_regression(
    'CPI',
    'HDI',
    method='log'
).encode()

curve_int + points_int
```



```
In [9]: alt.renderers.enable('notebook')
```

Out[9]: RendererRegistry.enable('notebook')

Final

Eu fiz outra visualização em que resolvi separa por continente e fazer uma analise por país dentro de sua região, continente e do mundo. Como ela deu 5 graficos diferentes, um para cada continente, resolvi não colocar-la aqui e a coloquei juntamente com os graficos feitos aqui, em:

- <https://carcesar.github.io/Visualizacao-T2/>

A nova visualização:

- Seleciona país pela legenda
- Seleciona região do continente
- Tooltips
- Retas de regressão continente e mundo para ter uma base de comparação

Pontos Importantes:

- o arquivo Rascunho.ipynb esta com as visualizações intermediárias relevantes. A ultima visualização do site tambem está lá.
- o arquivo Construção Data.ipynb foi onde construi os dados usados. Ele está na pasta Dados, junto com todos os '.csv's usados