

# **Conceção de modelos de aprendizagem e decisão**

Trabalho Prático - ADI 2024

Universidade do Minho

Diogo Abreu, Luís Figueiredo, Miguel Gramoso, e Rodrigo Monteiro  
{a100646, a100549, a100835, a100706}@alunos.uminho.pt

Grupo 21

## **1. Introdução**

Este relatório apresenta uma descrição do trabalho prático desenvolvido no âmbito da unidade curricular Aprendizagem e Decisão Inteligentes, cujo objetivo foi conceber e desenvolver um projeto utilizando os modelos de aprendizagem abordados ao longo do semestre.

## **2. Tarefa: *Dataset Grupo***

Esta tarefa consiste na consulta, análise e seleção de um dataset, o qual é explorado, analisado e preparado de modo a que sejam concebidos, otimizados e avaliados diversos modelos de *machine learning*.

Assim, a nossa pesquisa inicial levou-nos a escolher o *dataset* “World Values Survey”[1] – mais especificamente, o “Timeseries (1981 - 2022)”.

“The data can be used freely for non-commercial purposes such as research, publication, teaching etc.”

— <https://www.worldvaluessurvey.org/WVSDocumentationWVL.jsp>

O WVS consiste em pesquisas nacionalmente representativas realizadas em quase 100 países, que abrangem quase 90 por cento da população mundial, utilizando um questionário com inúmeras questões – que podem variar de acordo com o ano.

Escolhemos este *dataset*, dado que é a maior investigação transnacional, não comercial, em série temporal, sobre crenças e valores humanos alguma vez realizada, fornecendo assim uma grande quantidade de dados significativos. Estes dados são uma “impressão digital” dos valores humanos nos últimos 40 anos, uma visão global formada por descrições de mentalidades individuais, o que nos permite encontrar padrões e correlações entre ideias, valores e crenças.

A metodologia de análise de dados utilizada foi o CRISP-DM (“Cross-industry standard process for data mining”), excluindo a última etapa, uma vez que o modelo não será implementado ou monitorizado.

### 2.1. Estudo do negócio

Durante o estudo dos documentos de *code variables* dos diversos questionários, reparamos que diversos dados avaliam a religiosidade, – o que faz sentido, dado estar evidentemente relacionada com crenças e valores (tal como é exemplificado na imagem abaixo).

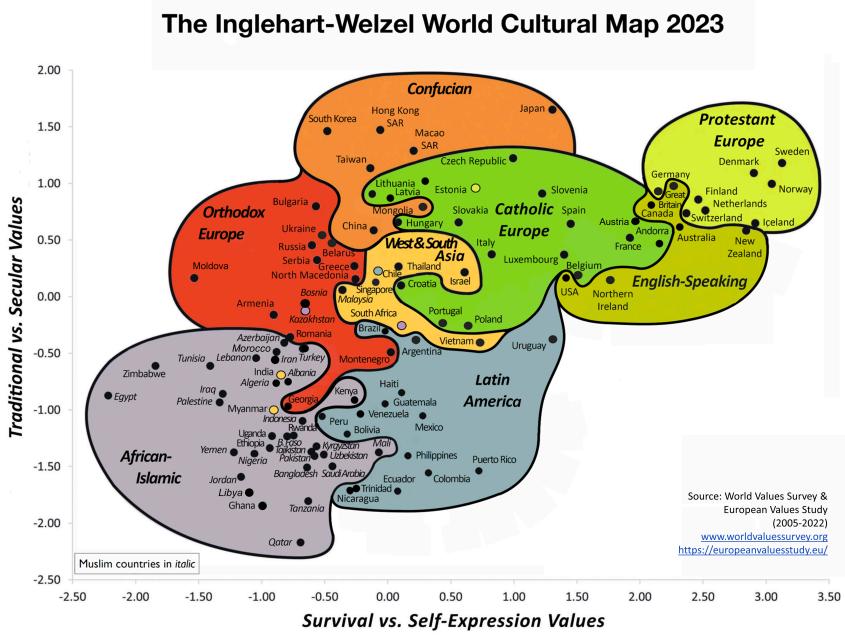


Figura 1: World Cultural Map

Portanto, decidimos tentar responder a esta questão:

Podemos utilizar *machine learning* para classificar pessoas como religiosas (e identificar a religião) ou não religiosas utilizando dados do WVS?

Reconhecemos que este tema requer alguma reflexão acerca de: O que é religião? Qual é a sua aparência a nível individual e coletivo? De que perspetiva devemos abordar estas questões? (sociológica, filosófica, psicológica, histórica, teológica, etc.).

Dada a natureza deste projeto, achamos mais relevante abordar apenas a nível sociológico (de forma superficial), dado que tanto *machine learning* como sociologia partilham um vínculo com análises estatísticas [2].

Não obstante, apresentamos, em anexo alguns pontos dessas perspetivas (Tabela 3).

Dada a grande quantidade de variáveis (mais de 1000 colunas), foi necessário selecionar um conjunto bastante mais pequeno. Para isso, usamos uma abordagem sociológica, tal como foi referido anteriormente, e escolhemos apenas variáveis que estivessem presentes em todos os questionários (ao longo dos anos).

A teoria da secularização afirma que a religião, no seu sentido tradicional, está em declínio à medida que o mundo avança em direção à industrialização e ao crescimento económico. Assim, incluímos variáveis como Ano e País.

A teoria da escolha racional afirma que as decisões de indivíduos são baseadas em “cálculos” para otimizar resultados, utilidade e satisfação. Tendo em conta esta teoria, interesses e objetivos pessoais são os principais motores de decisão humana, e, portanto, os indivíduos decidem o seu nível de compromisso com a religião, considerando os custos e benefícios. Assim, incluímos variáveis como Satisfação e Liberdade.

A teoria da segurança existencial baseia-se no desejo humano de estabilidade e segurança, e, por isso, estuda as ameaças ambientais, sociais e económicas à segurança humana. Focando nos padrões de mudança social e estruturas políticas, esta teoria também argumenta que a falta de segurança financeira causa uma maior presença de religiosidade. Assim, incluímos variáveis como Nível de Rendimento, Emprego e Saúde.

A teoria da estratificação social diz que a desigualdade resulta da estratificação desigual de recursos com base em classe, raça e gênero que afeta educação, saúde, etc. Assim, incluímos variáveis como Sexo e Nível de Educação.

Portanto, o objetivo é investigar os padrões lineares e não lineares nos dados para identificar os principais indicadores de religiosidade no contexto global e prever a religiosidade.

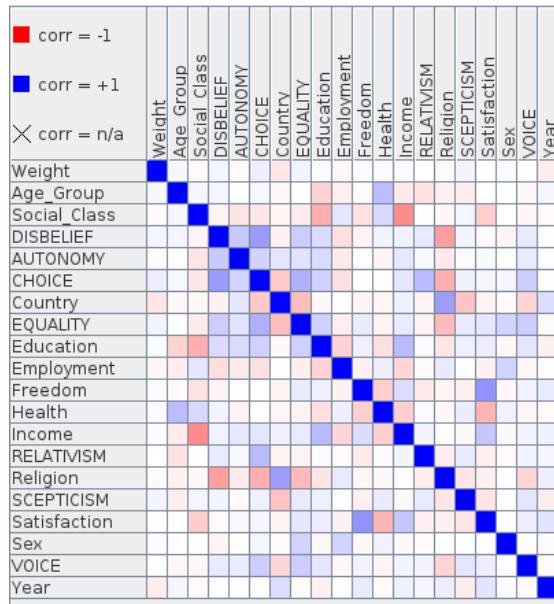
## 2.2. Estudo dos dados

O dataset utilizado contém 399 000 linhas e 20 colunas.

Para além das variáveis mencionadas anteriormente, também utilizamos alguns índices de Welzel (calculados a partir das respostas dadas no questionário).

Nome	Código	Informação adicional
Idade	X003	
Nível de educação	X025R	“recoded”
Saúde	A009	“state of health (subjective)”
Liberdade	A173	“How much freedom of choice and control”
Nível de rendimento	X047_WVS	“Scale of incomes”
Religião	F025	“Religious denominations - major groups”
Satisfação	A170	“Satisfaction with your life”
Emprego	X028	“Employment status”
Sexo	X001	
Ano	S020	“Year survey”
País	COW_ALPHA	“CoW country code alpha”
<i>Autonomy</i>	Y021	“Wezel Autonomy subindex”
<i>Choice</i>	Y023	“Welzel choice sub-index”
<i>Equality</i>	Y022	“Welzel equality sub-index”
<i>Relativism</i>	Y013	“Welzel relativism”
<i>Scepticism</i>	Y014	“Welzel scepticism index”
<i>Voice</i>	Y024	“Welzel voice sub-index”
<i>Weight</i>	S018	“Equilibrated weight-1000”

Tabela 1: Variáveis selecionadas

Figura 2: Rank Correlation - *Dataset Grupo*<sup>1</sup>

### 2.2.1. Religiosidade por país

**Religiosity World Map**

World map of the selected countries and their average religiosity score

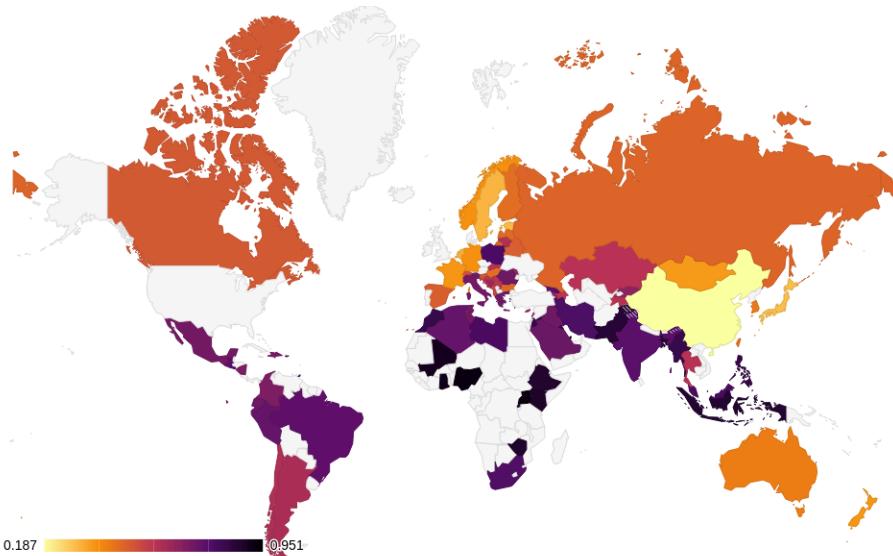


Figura 3: Religiosidade por país

Este mapa foi obtido com recurso a um script em python e ao componente Choropleth Map [3].

---

<sup>1</sup>A variável “Classe Social” foi retirada devido à alta correlação com o nível de rendimento

O script em python transforma os códigos dos países no nome completo de modo a que o output seja reconhecido pelo componente que gera o mapa.

```
import knime.scripting.io as knio

rows = knio.input_tables[0].to_pandas()

for index, row in rows.iterrows():
    alpha_code = row['Country']
    country_name = alpha_to_name.get(alpha_code, "Unknown")
    rows.loc[index, 'Country'] = country_name

knio.output_tables[0] = knio.Table.from_pandas(rows)
```

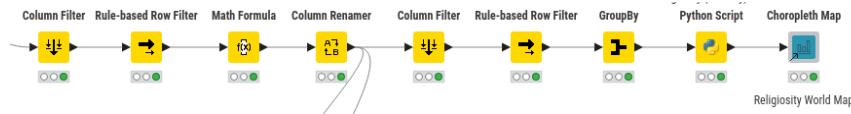


Figura 4: Religiosidade por país - KNIME workflow

### 2.2.2. Religiosidade por ano

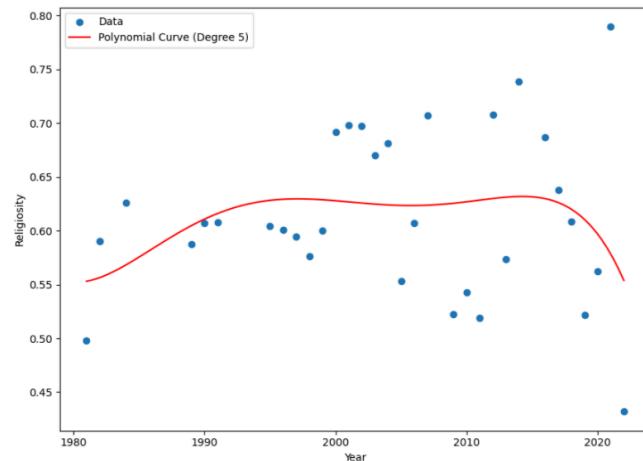


Figura 5: Religiosidade por ano (global)

Achamos importante salientar que é preciso ter prudência ao tirar conclusões a partir de dados estatísticos. O gráfico pode indicar vagamente (com uma grande margem de erro) que a religiosidade está estável ou talvez a diminuir ao longo dos anos. No entanto, essa generalização não tem muito valor, dado que esconde a diversidade dos subgrupos [4]. O seguinte gráfico mostra a religiosidade ao longo dos anos na Espanha:

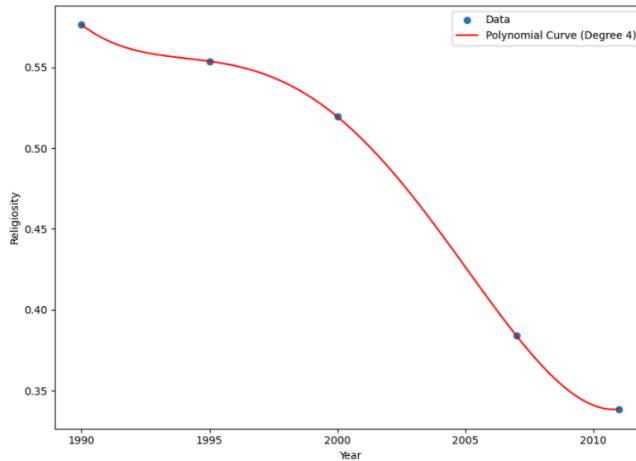


Figura 6: Religiosidade por ano (Espanha)

Para além disso, também é importante salientar que nem todos os anos têm a mesma quantidade de dados. Para isso, seria necessário usar *equal size sampling*, mas sobre o custo da perda de dados.

Para obter este tipo de gráfico, utilizamos o nodo Python View, visto que queríamos utilizar o método dos mínimos quadrados para ajustar uma curva polinomial aos dados.

```

data = knio.input_tables[0].to_pandas()
x_values = data.iloc[:, 0]
y_values = data.iloc[:, 1]

degree = 4
coefficients = np.polyfit(x_values, y_values, degree)
poly = np.poly1d(coefficients)

plt.scatter(x_values, y_values, label='Data')
x_curve = np.linspace(min(x_values), max(x_values), 100)

plt.plot(
    x_curve, poly(x_curve), color='red',
    label=f'Polynomial Curve (Degree {degree})'
)

plt.xlabel('Year')
plt.ylabel('Religiosity')

buffer = BytesIO()
plt.savefig(buffer, format='png')
buffer.seek(0)
knio.output_view = knio.view(buffer.getvalue())

```

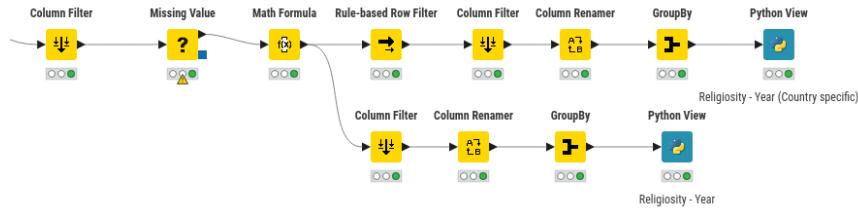


Figura 7: Religiosidade por ano - KNIME workflow

### 2.2.3. Religiosidade por idade

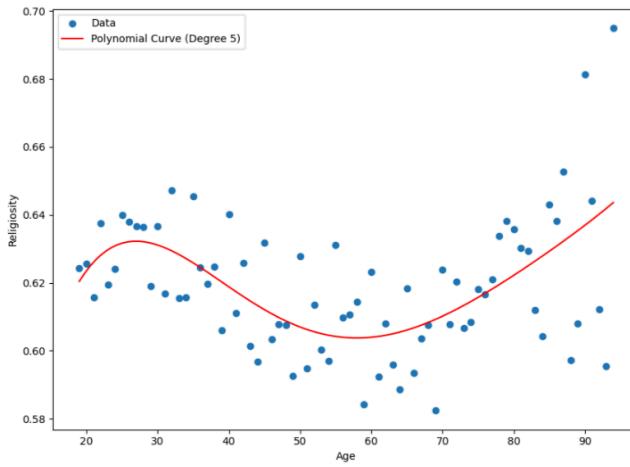


Figura 8: Religiosidade por idade

### 2.2.4. Religiosidade por nível de educação

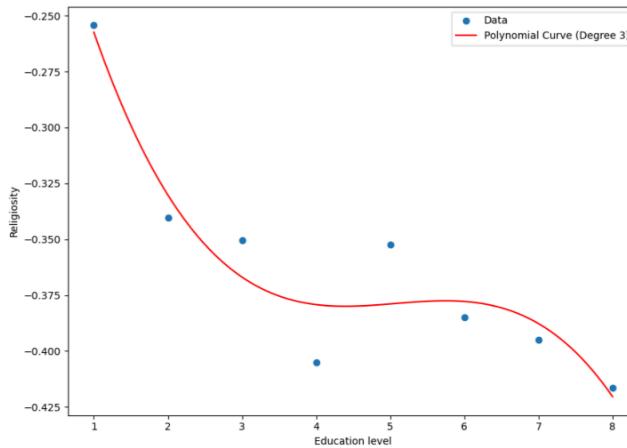


Figura 9: Religiosidade por nível de educação

### 2.2.5. Religiosidade por nível de rendimento

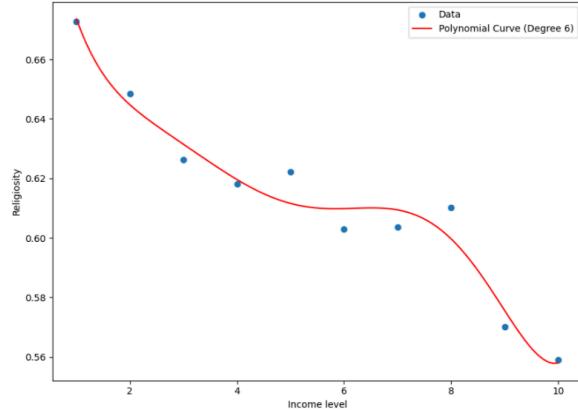


Figura 10: Religiosidade por nível de rendimento (global)

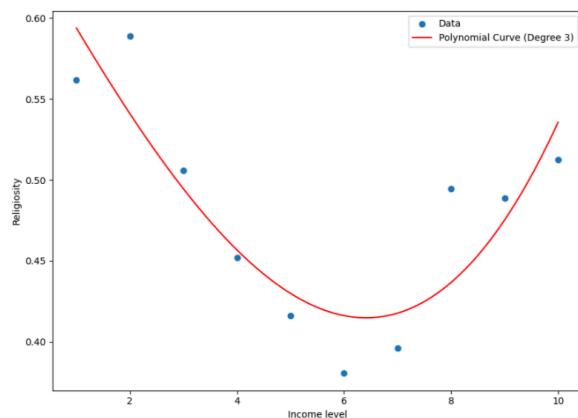


Figura 11: Religiosidade por nível de rendimento (Espanha)

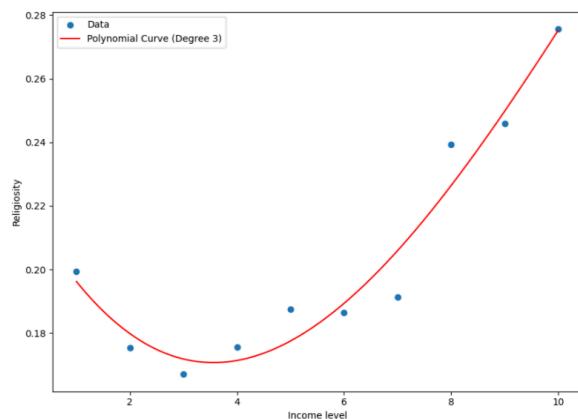


Figura 12: Religiosidade por nível de rendimento (China)

### 2.2.6. Outros<sup>2</sup>

Ver em anexo:

- Religiosidade por nível de satisfação: Figura 52
- Most important child qualities: Figura 53,
- Justifiable: Figura 54,
- “Ever felt ...”: Figura 55,
- Memberships: Figura 56
- “I see myself as someone who ...”: Figura 57
- Most important for successful marriage: Figura 58
- Equality world map: Figura 60
- Scepticism world map: Figura 61
- Satisfaction world map: Figura 62

## 2.3. Preparação dos dados

### 2.3.1. Seleção e tratamento de colunas

- Foi utilizado o nodo Column Filter para remover todas as colunas que não foram selecionadas no estudo dos dados.
- Para a substituição de valores negativos (equivalentes à falta de uma resposta) por *missing values* foram utilizados os nodos Column Expressions, por ser *multi-column*, e o nodo Rule-based Row Filter.
- Para remover colunas com mais de 15% de *missing values* foi utilizado o nodo Missing Value Column Filter. Umas das variáveis selecionadas inicialmente teve de ser retirada (X051 - Ethnic group)
- Para renomear e reordenar colunas foram utilizados os nodos Column Renamer e Column Resorter

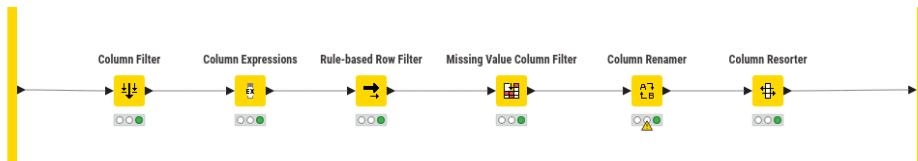


Figura 13: Seleção e tratamento de colunas - KNIME workflow

### 2.3.2. Tratamento de outliers e *missing values*

- Foi utilizado o nodo Numeric Outliers, que substitui os outliers pelos valores mais próximos permitidos:

<sup>2</sup>Variáveis não utilizadas na modelação foram usadas na criação dos gráficos de barras

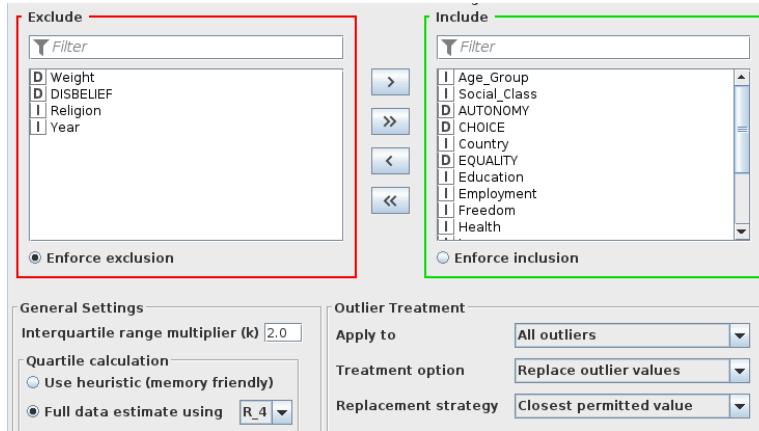


Figura 14: Exemplo do tratamento de outliers - Dataset Grupo

- Foi necessário ordenar as linhas por ano e, em caso de empate, por país (no caso de mais empates, esta é a restante ordem de ordenação: idade, sexo, relativismo)
- Os *missing values* são substituídos pelo valor da *moving average* [5]. Achamos que esta é uma boa abordagem, assumindo que um qualquer indivíduo tenha, geralmente, mais em comum com alguém que respondeu no mesmo ano (ou num ano próximo) e que talvez seja do mesmo país, do que o contrário.
- No entanto, este tratamento pode deixar passar alguns *missing values*, por isso, aplicamos novamente o nodo mas com a opção *remove row*.

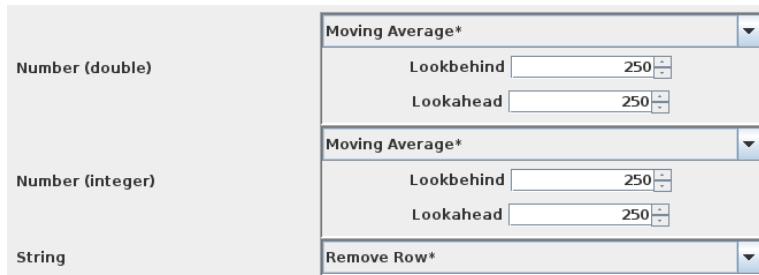


Figura 15: Exemplo do tratamento de missing values - Dataset Grupo

Figura 16: Tratamento de outliers e *missing values* - KNIME workflow

### 2.3.3. Agrupamento de valores

- Foi utilizado o nodo Java Snippet para agrupar religiões: achamos melhor agrupar as religiões do cristianismo dadas as suas semelhanças (Roman Catholic, Protestant, Orthodox, other)

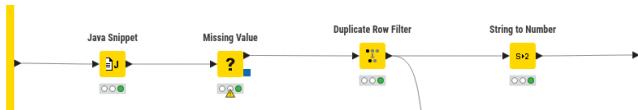


Figura 17: Tratamento de outliers e *missing values* - KNIME workflow

### 2.3.4. Oversampling das classes em minoria

- Foi utilizado o nodo SMOTE (Synthetic Minority Oversampling Technique) para criar mais instâncias das classes em minoria. (Para poupar tempo, aplicamos o SMOTE apenas em 10% dos dados)

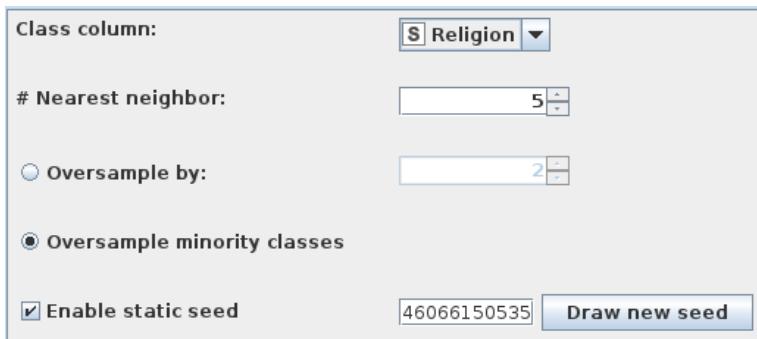


Figura 18: SMOTE - oversampling minority classes

---

Consider a sample (6,4) and let (4,3) be its nearest neighbor.  
(6,4) is the sample for which k-nearest neighbors are being identified.  
(4,3) is one of its k-nearest neighbors.  
Let:  
 $f_{1,1} = 6 \quad f_{2,1} = 4 \quad f_{2,1} - f_{1,1} = -2$   
 $f_{1,2} = 4 \quad f_{2,2} = 3 \quad f_{2,2} - f_{1,2} = -1$   
The new samples will be generated as  
 $(f'_1, f'_2) = (6,4) + \text{rand}(0-1) * (-2,-1)$   
rand(0-1) generates a random number between 0 and 1.

---

Figura 19: Exemplo da geração de *samples* sintéticos [6]

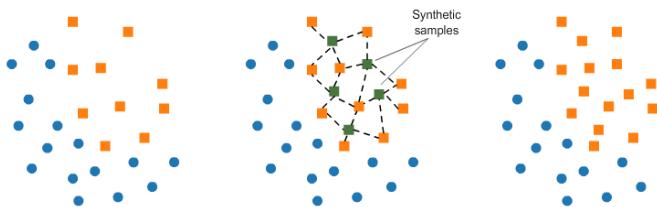


Figura 20: *Oversampling* através de SMOTE [7]

### 2.3.5. Undersampling

- Dada a grande quantidade de dados, achamos benéfica a aplicação de um algoritmo de *undersampling*
- Como não encontramos nenhum nodo *default* do KNIME, ou componente no *KNIME Hub* que fizesse *undersampling*, decidimos fazê-lo através de um script
- Para isso, utilizamos o nodo Conda Environment Propagation [8] de modo a conseguirmos utilizar os *packages* necessários para o nodo Python Script (legacy) que aplica a técnica Tomek Links [9]

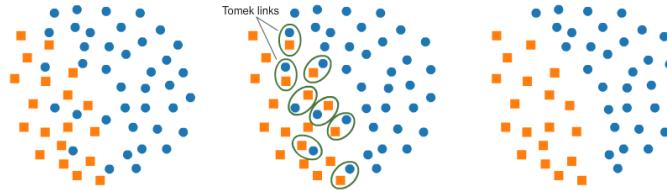


Figura 21: *Undersampling* através de Tomek Links [7]

```
from imblearn.under_sampling import TomekLinks
from sklearn.datasets import make_classification

input_table_1_cleaned = input_table_1.dropna()
religion_column = input_table_1_cleaned['Religion']
other = input_table_1_cleaned.drop('Religion', axis=1)

tl = TomekLinks()
X_res, y_res = tl.fit_resample(other, religion_column)

output_table_1 = pd.concat([pd.DataFrame(X_res, columns=other.columns),
                           pd.Series(y_res, name='Religion')], axis=1)
```

### 2.4. Modelação com XGBoost Tree Ensemble

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	4 (Predicted)	5 (Predicted)			
0 (Actual)	35696	19612	698	2029	1357	3664	56.61%		
1 (Actual)	7502	124333	487	3124	2073	1210	89.62%		
2 (Actual)	626	683	18607	204	675	247	88.43%		
3 (Actual)	730	4286	263	73249	1311	843	90.79%		
4 (Actual)	721	912	174	1238	25882	807	87.05%		
5 (Actual)	4211	616	120	638	530	26237	81.10%		
	72.13%	82.65%	91.44%	91.01%	81.32%	79.49%			
Overall Statistics									
Overall Accuracy	83.15%	Overall Error	16.85%	Cohen's kappa ( $\kappa$ )	0.776	Correctly Classified	304004	Incorrectly Classified	61591

Figura 22: Resultados do modelo XGBoost Tree Ensemble com cross-validation, oversampling, e undersampling

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	4 (Predicted)	5 (Predicted)
0 (Actual)	42865	32093	596	3345	694	6279
1 (Actual)	12044	158758	739	5339	740	5828
2 (Actual)	277	801	17416	229	52	443
3 (Actual)	939	7315	342	86723	1576	1241
4 (Actual)	275	1380	52	1305	25830	646
5 (Actual)	6018	1040	89	990	327	24321
68.67%		78.83%	90.55%	88.56%	88.40%	62.75%
Overall Statistics						
Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified		
79.28%	20.72%	0.716	355913	93034		

Figura 23: Resultados do modelo XGBoost Tree Ensemble com cross-validation e oversampling

	0.0 (Predicted)	1.0 (Predicted)	2.0 (Predicted)	3.0 (Predicted)	4.0 (Predicted)	5.0 (Predicted)
0.0 (Actual)	30828	22850	411	2624	617	2211
1.0 (Actual)	8678	147544	999	5563	1500	2000
2.0 (Actual)	118	687	1009	109	26	11
3.0 (Actual)	640	5261	176	75543	843	576
4.0 (Actual)	116	762	27	658	9173	113
5.0 (Actual)	3877	726	29	774	125	7386
69.66%		82.97%	38.06%	88.59%	74.67%	60.06%
Overall Statistics						
Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified		
81.14%	18.86%	0.708	271483	63107		

Figura 24: Resultados do modelo XGBoost Tree Ensemble com cross-validation e undersampling

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	4 (Predicted)	5 (Predicted)
0 (Actual)	35008	31653	1150	3024	981	3285
1 (Actual)	13185	157129	1860	5633	2155	3559
2 (Actual)	140	686	976	112	27	21
3 (Actual)	817	7214	327	78419	1374	617
4 (Actual)	213	1273	50	1031	9837	171
5 (Actual)	5471	1160	86	1004	171	8358
63.84%		78.91%	21.94%	87.89%	67.63%	52.20%
Overall Statistics						
Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified		
76.61%	23.39%	0.644	289727	88450		

Figura 25: Resultados do modelo XGBoost Tree Ensemble com cross-validation (sem oversampling e undersampling)

Assim, verifica-se que a melhor opção é o uso de *oversampling* e de *undersampling*, uma vez que aumenta a *accuracy* em 6.31%, a sensibilidade da categoria 5 em 23.74%, a sensibilidade da categoria 2 em 38.9%, a especificidade da categoria 5 em 27,85% etc.

### 2.5. Comparação com previsão humana

De modo a comparar o desempenho do modelo com o desempenho humano, fizemos um questionário com 20 questões, cada uma com os mesmos dados que o modelo utilizou para fazer a previsão – as linhas correspondentes foram recolhidas aleatoriamente do dataset.

Com uma pequena amostra de respostas, obtivemos uma média de 8.8/20 – menor do que a pontuação do modelo de 17/20.

Age: Menos de 34 anos; * Education level: 2/3; Employment status: Full time; Freedom: 5/10; Health: Good; Income level: 5/10; Satisfaction with life: 8/10; Sex: Female; Year: 2022; Country: Czech Republic	1 ponto
Autonomy Index: 0.33; Choice Index: 0.85; Equality Index: 0.80; Relativism Index: 1; Scepticism Index: 0.33; Voice Index: 0.41;	
<input type="radio"/> Christian (Roman Catholic, Protestant, Orthodox, other) <input type="radio"/> Jew <input type="radio"/> Muslim <input type="radio"/> Hindu <input type="radio"/> Buddhist <input type="radio"/> None	

#### Estatísticas



### 3. Tarefa: Dataset Atribuído

#### 3.1. Estudo dos dados

O conjunto de dados contém valores laboratoriais de doadores de sangue e pacientes com hepatite C, além de valores demográficos como idade.

Nome	Tipo	Coluna
ID	Integer	id
Age	Integer	age
Birth year	Integer	year_of_birth
Birth month	Integer	moth_of_birth
Birth day	Integer	day_of_birth
Sex	String	Sex
Birth location	String	birth_location
Albumin	String	ALB
Alkaline phosphatase	String	ALP
Alanine transferase	String	ALT
Aspartate transferase	String	AST
Bilirubin	String	BIL
Cholinesterase	String	CHE
Cholesterol	String	CHOL
Creatinine	String	CREA
Gama Glutamil Transferase	String	CGT
Protein	String	PROT
Category (target variable)	String	Category

Tabela 2: Variáveis

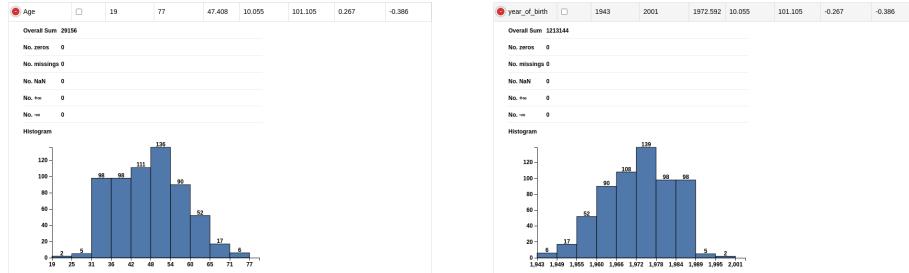


Figura 28: Comparação dos histogramas das variáveis Age e year\_of\_birth

Assumindo que os dados foram recolhidos aproximadamente no mesmo período de tempo, então pode-se retirar os dados acerca do ano de nascimento, visto que estes terão essencialmente a mesma informação que os dados da variável idade.

Tal verifica-se dada a correlação entre as duas variáveis (verifica-se também que as variáveis `month_of_birth` e `day_of_birth` são pouco significativas):

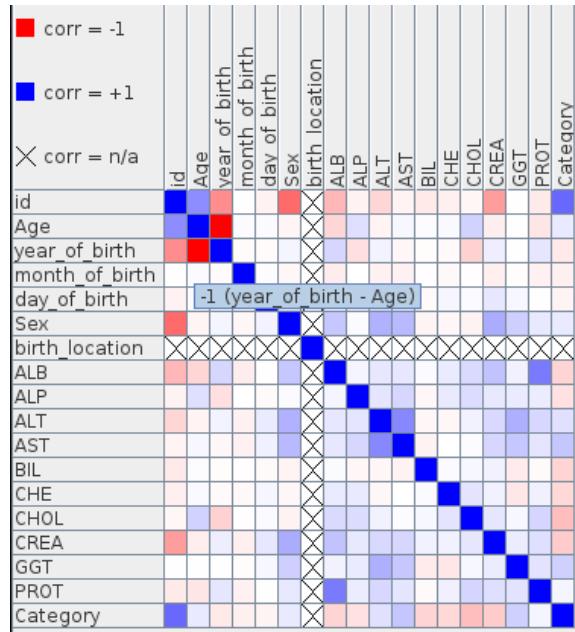


Figura 29: Rank correlation - *Dataset Atribuído*

Para além disso, também é possível verificar através da exploração de dados que a variável `Sex` possui dois valores, quando deveria ser binária. Portanto, os valores “mm” têm de ser corrigidos para apenas “m”. Também se verifica na coluna `birth_location` existe apenas um valor único, logo essa coluna não oferece informação ao modelo e pode ser retirada.

Column	Exclude Column	No. missings	Unique values	All nominal values	Frequency Bar Chart
Sex	<input type="checkbox"/>	0	3	m, f, mm	
birth_location	<input type="checkbox"/>	44	1	New Delhi	

Figura 30: Estatísticas sobre as variáveis `Sex` e `birth_location`

Como se pode notar na figura abaixo, será necessário fazer tratamento de outliers em algumas variáveis (como ALP, ALT, AST, BL, CREA e GGT).

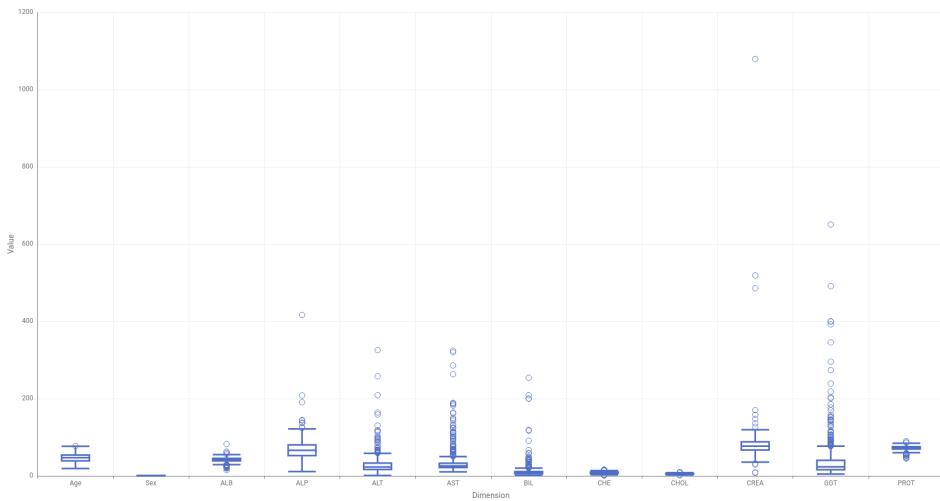
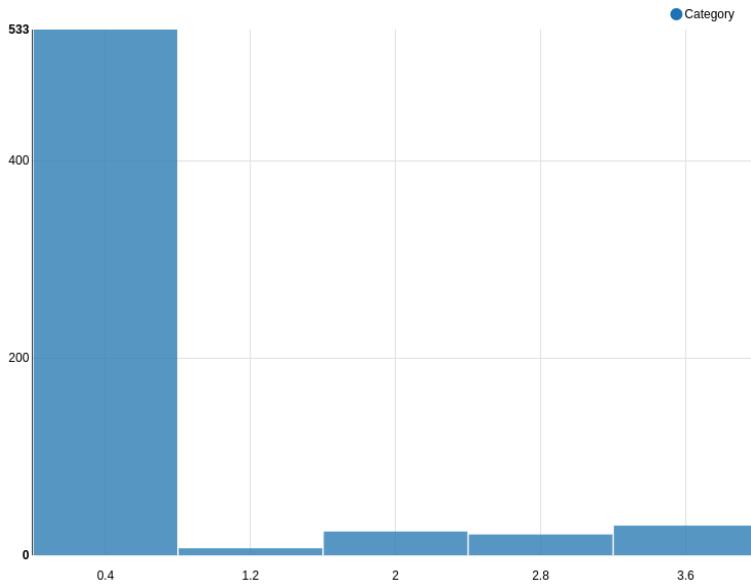


Figura 31: Outliers - Dataset atribuído

Por fim, verificamos que existe um desbalanceamento dos dados relativamente à variável **Category**:



Dada a pequena quantidade de dados da categoria “0=suspect Blood Donor” e a sua relação com a categoria “0=Blood Donor”, determinamos que estas duas categorias poderiam ser agrupadas numa só – dito de uma maneira informal, é estranho prever se se suspeita de uma categoria, quando qualquer categoria é suspeita quando há a necessidade de prever. A partir deste histograma, também consideramos que a utilização de uma técnica de *oversampling* (para as categorias em minoria) seria útil para a otimização dos modelos criados.

### 3.2. Preparação dos dados

1. Filtragem de colunas irrelevantes através do nodo Column Filter
  - Colunas retiradas: `id`, `year_of_birth`, `month_of_birth`, `day_of_birth`, `birth_location`
2. Tratamento das colunas `Sex` e `Category` através do nodo String Replacer
  - Padrão regex para a coluna `Category`: `(\d)(s?)\=(?:.+)`
  - Padrão regex para a coluna `Sex`: `(m|s){2,}`
3. Transformar o tipo de algumas colunas para `double` através do nodo String to Number (como consequência, substitui a string “NA” por *missing value*)
4. Tratamento de *missing values*
  - Ordenar os dados de acordo com idade, e em caso de empate, de acordo com o sexo através do nodo Sorter
  - Substituir números em falta pela *moving average* [5] através do nodo Missing Values
5. Tratamento de outliers através do nodo Numeric Outliers
  - Os outliers são substituídos pelo valor mais próximo permitido

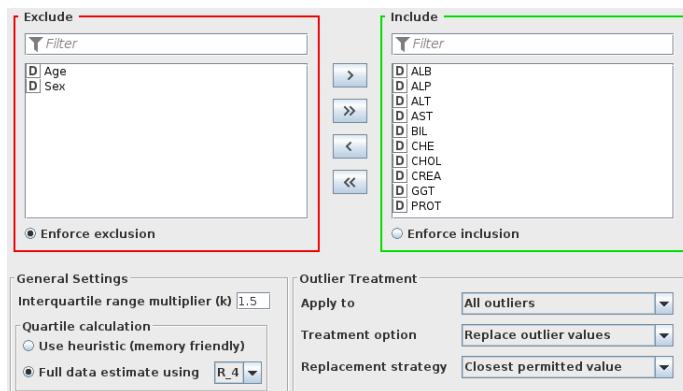
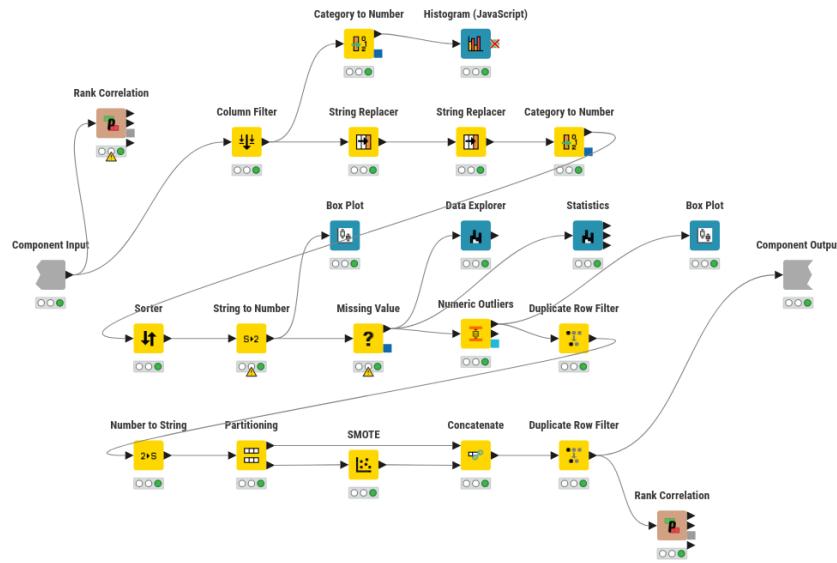


Figura 33: Configurações do nodo Numeric Outliers

6. *Oversampling* das classes em minoria
  - Aplicar o nodo SMOTE em 30% dos dados com *stratified sampling* configurado com a opção `oversample minority classes`

Figura 34: Preparação de dados - *Dataset atribuído*

### 3.3. Modelação

Nos diversos modelos foi utilizada a técnica *cross-validation* (nodos X-Partitioner e X-Aggregator) de modo a efetuar avaliações com diferentes grupos de treino e de teste e evitar enviesamento na seleção dos grupos.

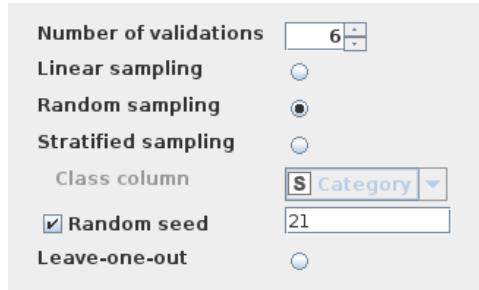
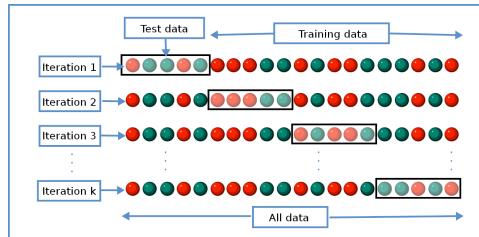


Figura 35: Configuração do nodo X-Partitioner

Figura 36: Visualização de *cross-validation*

### 3.3.1. Modelação sem *oversampling*

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	
0 (Actual)	538	0	2	0	99.63%
1 (Actual)	10	5	7	2	20.83%
2 (Actual)	5	7	7	2	33.33%
3 (Actual)	1	2	5	22	73.33%
	97.11%	35.71%	33.33%	84.62%	

Overall Statistics					
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified	
93.01%	6.99%	0.659	572	43	

Figura 37: Resultados de Gradient Boosted Trees sem *oversampling*

Apesar de se verificar uma *accuracy* alta (93%), também se verifica uma sensibilidade baixa para as categorias diferentes de “0” (21%, 33%, 73%). No caso de desbalanceamento de dados, a *accuracy* pode não ser a melhor medida de avaliação do desempenho do modelo [10], sendo, por isso, necessário aplicar técnicas de *oversampling*.

### 3.3.2. Modelação com *oversampling*

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	
0 (Actual)	534	2	2	2	98.89%
1 (Actual)	8	168	2	1	93.85%
2 (Actual)	5	6	163	3	92.09%
3 (Actual)	1	1	0	180	98.90%
	97.45%	94.92%	97.60%	96.77%	

Overall Statistics					
Overall Accuracy	Overall Error	Cohen's kappa ( $\kappa$ )	Correctly Classified	Incorrectly Classified	
96.94%	3.06%	0.954	1045	33	

Figura 38: Resultados de Gradient Boosted Trees com *oversampling*

Com *oversampling*, não só a *accuracy* aumentou para 97%, mas os valores de sensibilidade também aumentaram todos para mais de 90% – uma subida de 70% para a categoria “1”.

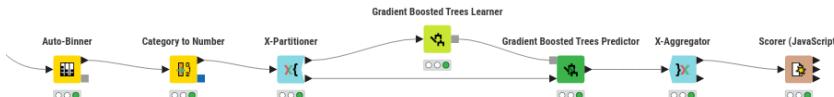
#### 3.3.2.1. Modelação com dados normalizados (*z-score normalization*)



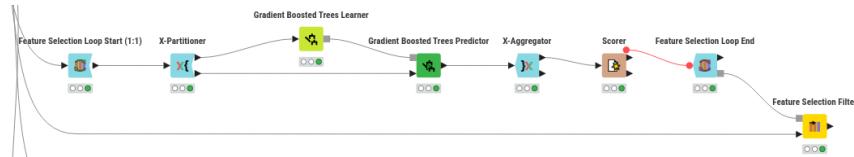
Figura 39: Modelação com dados normalizados - workflow KNIME

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	
0 (Actual)	530	6	3	1	98.15%
1 (Actual)	4	165	8	2	92.18%
2 (Actual)	3	5	166	2	94.32%
3 (Actual)	3	0	3	175	96.69%
98.15%		93.75%	92.22%	97.22%	
Overall Statistics					
Overall Accuracy	96.28%	Overall Error	3.72%	Cohen's kappa ( $\kappa$ )	0.944
Correctly Classified	1036	Incorrectly Classified	40		

Figura 40: Resultados de Gradient Boosted Trees com normalização dos dados

3.3.2.2. Modelação com dados *binned*Figura 41: Modelação com dados *binner* - workflow KNIME

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)	
0 (Actual)	526	3	4	7	97.41%
1 (Actual)	10	165	3	1	92.18%
2 (Actual)	8	1	165	3	93.22%
3 (Actual)	10	2	1	169	92.86%
94.95%		96.49%	95.38%	93.89%	
Overall Statistics					
Overall Accuracy	95.08%	Overall Error	4.92%	Cohen's kappa ( $\kappa$ )	0.926
Correctly Classified	1025	Incorrectly Classified	53		

Figura 42: Resultados de Gradient Boosted Trees com dados *binned*3.3.2.3. Modelação com *feature selection*Figura 43: Modelação com *feature selection* - workflow KNIME

Optimization Criterion: The score is being maximized.	
Accuracy	Nr. of features
0,968	11
0,968	10
0,966	12
0,966	11
0,966	10
0,962	11
0,96	10
0,958	8
0,957	9
0,955	6
0,955	5
0,952	11
0,952	7
0,951	9
0,951	8
0,95	9
0,95	8
0,95	7
0,946	9
0,945	10
0,944	8
0,942	6
0,941	6
0,94	6
0,94	6
0,937	7
0,919	6
0,907	5

Figura 44: Resultados da *feature selection*

Verifica-se que é melhor manter todas a variáveis selecionadas, ou maior parte, tendo como medida a *accuracy*, visto que a diferença entre os valores do topo não é significativa.

### 3.3.2.4. Modelação com regressão

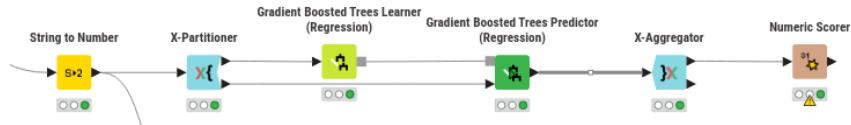
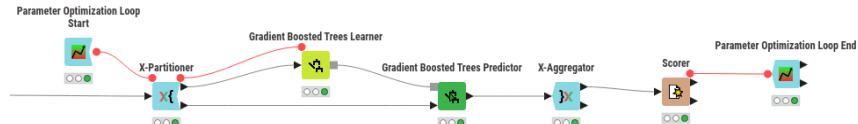


Figura 45: Modelação com regressão - workflow KNIME

R <sup>2</sup> :	0,938
Mean absolute error:	0,122
Mean squared error:	0,083
Root mean squared error:	0,288
Mean signed difference:	-0,005
Mean absolute percentage error:	NaN
Adjusted R <sup>2</sup> :	0,938

Figura 46: Resultados de Gradient Boosted Trees Learner (regression)

### 3.3.2.5. Modelação com otimização de parâmetros

Figura 47: Modelação com *parameter optimization* - workflow KNIME

	0 (Predicted)	1 (Predicted)	2 (Predicted)	3 (Predicted)					
0 (Actual)	535	2	2	1	99.07%				
1 (Actual)	6	170	2	1	94.97%				
2 (Actual)	3	5	165	4	93.22%				
3 (Actual)	2	1	0	179	98.35%				
	97.99%	95.51%	97.63%	96.76%					
Overall Statistics									
Overall Accuracy	97.31%	Overall Error	2.69%	Cohen's kappa ( $\kappa$ )	0.959	Correctly Classified	1049	Incorrectly Classified	29

Figura 48: Resultados da otimização de parâmetros

#	RowID	treeDepth Number (integer)	nrModels Number (integer)	learningRate Number (double)
1	Best ...	3	90	0.352

Figura 49: Melhores parâmetros

Standard settings Flow Variables Job Manager Selection

Parameters

Parameter	Start value	Stop value	Step size	Integer?
treeDepth	1	15	1.0	<input checked="" type="checkbox"/>
nrModels	80	125	1.0	<input checked="" type="checkbox"/>
learningRate	0.1	0.4	0.1	<input type="checkbox"/>

+ Add new parameter

Strategy settings

Search strategy Bayesian Optimization (TPE)

Random seed 21

Enable step size

Max. number of iterations 200

Number of warm-up rounds 20

Gamma 0,25

Number of candidates per round 25

Figura 50: Configuração do nodo Parameter Optimization Loop Start

### 3.3.2.6. Visualização de uma *decision tree*

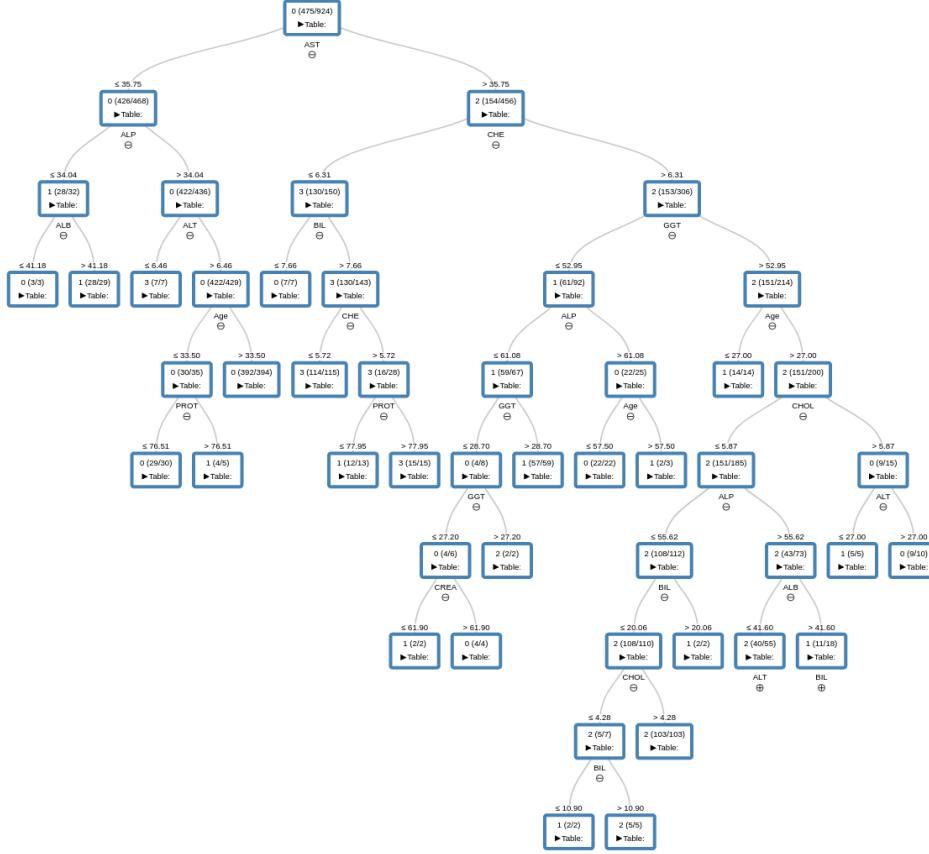


Figura 51: Decision Tree - *Dataset atribuído*

Nesta árvore de decisão (grafo hierarquizado), cada nodo interno testa um atributo, cada ramo identifica um intervalo de valores do nodo testado, e cada folha representa uma decisão. Assim, por exemplo, se uma linha tiver AST menor ou igual a 35.75, ALP menor ou igual a 34.04 e ALB maior do que 41.18, então prevê-se que é da categoria “1”.

## 4. Conclusões

Achamos que fomos capazes de aplicar vários conceitos associados à conceção de modelos de aprendizagem e decisão abordados ao longo do semestre e também conceitos não abordados, tal como *oversampling* e *undersampling*. Por fim, apreciamos a liberdade que nos foi dada na escolha de um dataset, uma vez que nos motivou mais a explorar os dados, a produzir estatísticas e a testar diferentes técnicas de modelação.

## 5. Anexos

Perspetiva	Tópicos
Sociológica	<ul style="list-style-type: none"> <li>– Teoria da escolha racional</li> <li>– Teoria da segurança existencial</li> <li>– Teoria da estratificação social</li> <li>– Teoria da secularização</li> </ul>
Filosófica	<ul style="list-style-type: none"> <li>– Argumentos ontológicos, cosmológicos, ..., para a existência de Deus</li> <li>– Diversidade e pluralismo</li> <li>– Deísmo, teísmo, ateísmo, gnosticismo, agnosticismo, etc.</li> <li>– Paradoxo da omnipotência</li> </ul>
Teológica	<ul style="list-style-type: none"> <li>– Estudo das Escrituras Sagradas</li> <li>– Problema do bem e do mal</li> </ul>
Psicológica	<ul style="list-style-type: none"> <li>– Modelo de James Fowler</li> <li>– Argumento eudemonológico</li> <li>– Interpretação Freudiana e Junguiana</li> <li>– Simbolismo e sincronicidade Junguiana</li> </ul>
Histórica	<ul style="list-style-type: none"> <li>– Religião e rituais no Antigo Egito</li> <li>– Religião na Antiga Mesopotâmia</li> <li>– A história primitiva de Yahweh</li> </ul>

Tabela 3: Alguns tópicos das diversas perspetivas sobre religião

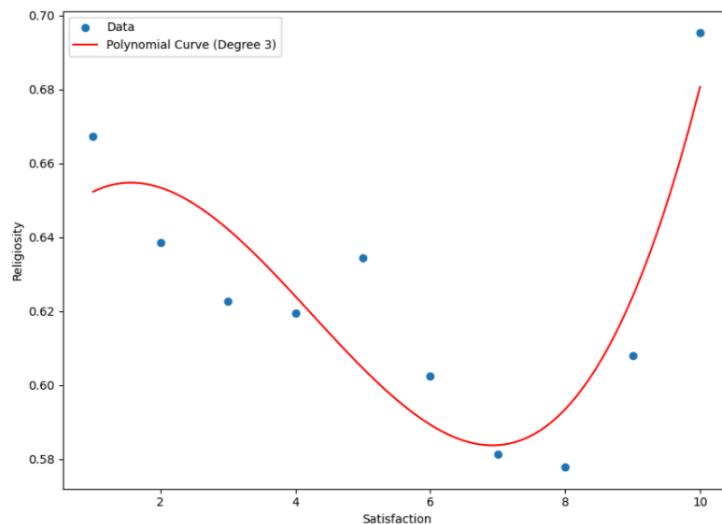
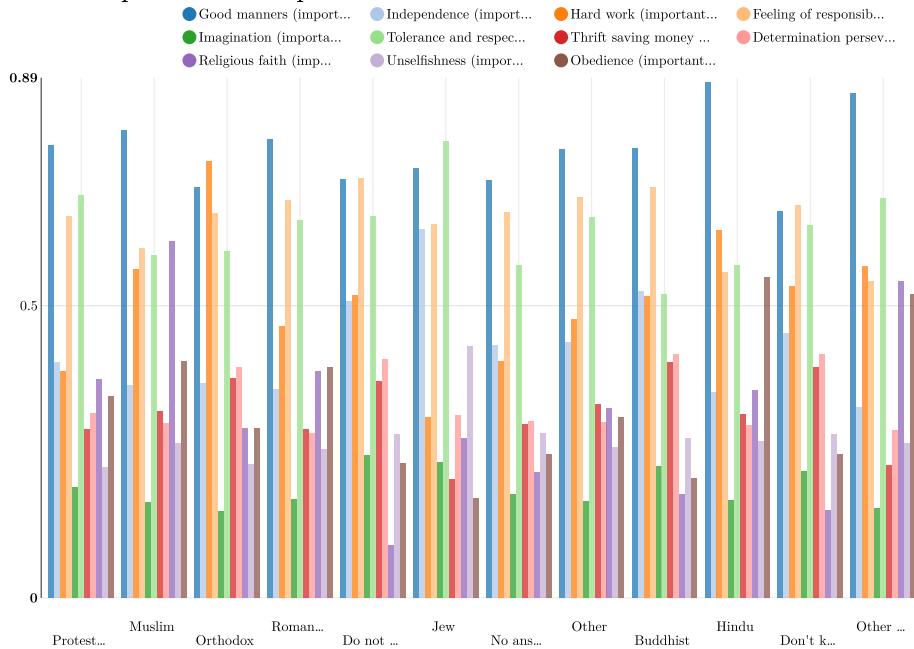
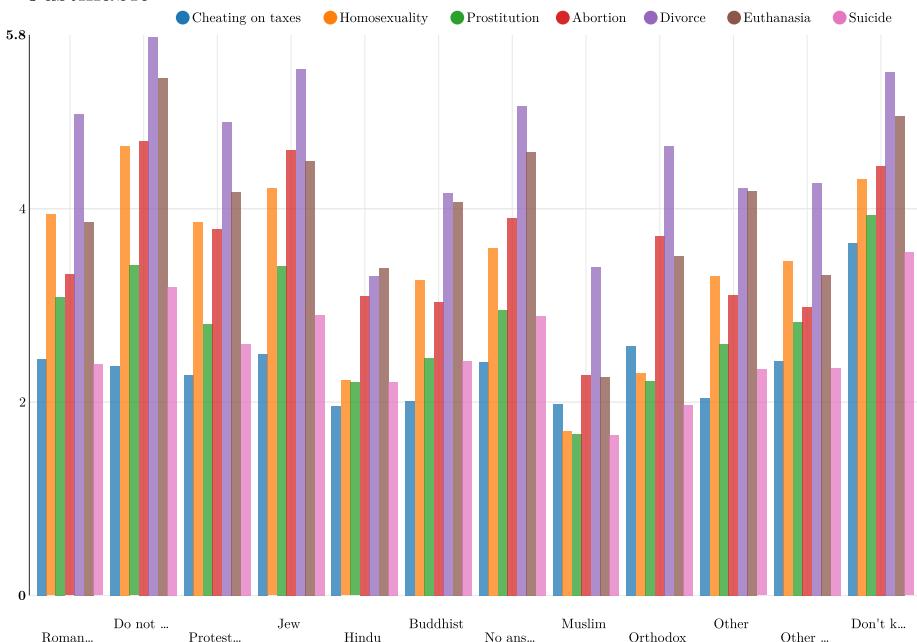


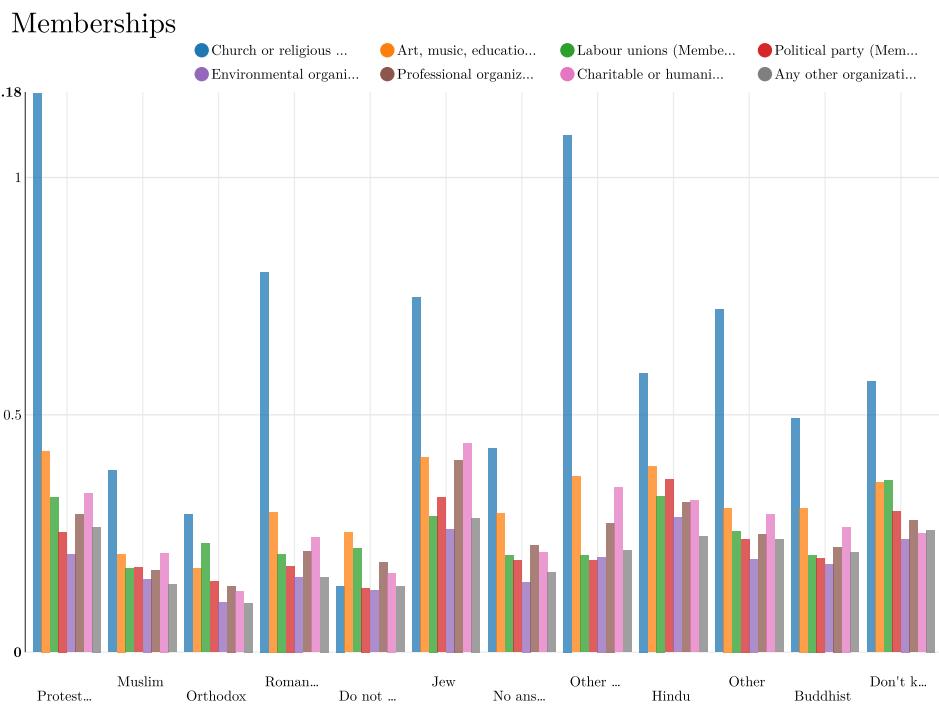
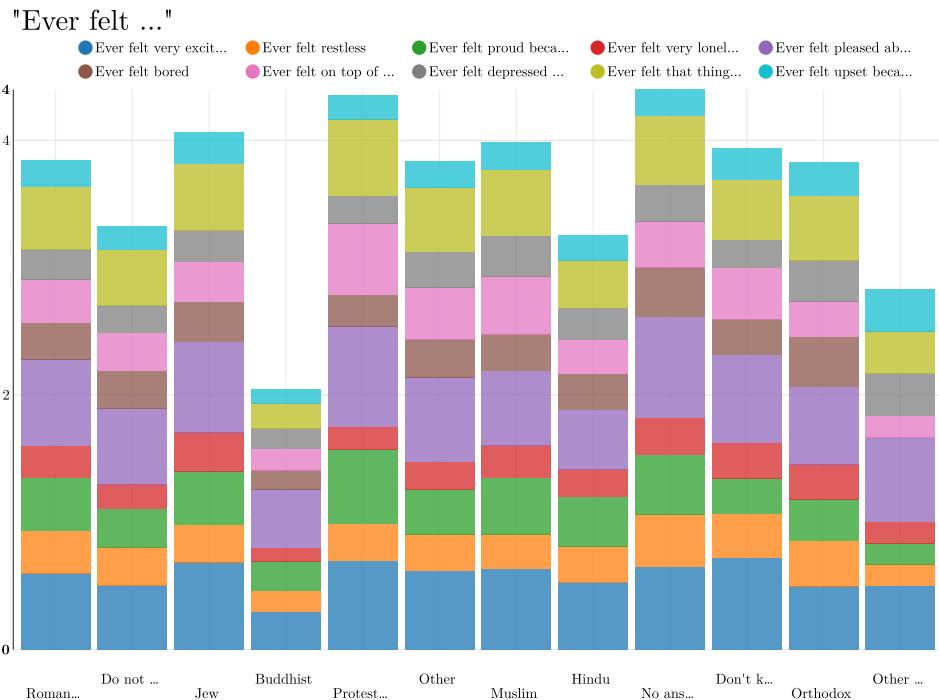
Figura 52: Religiosidade por nível de satisfação

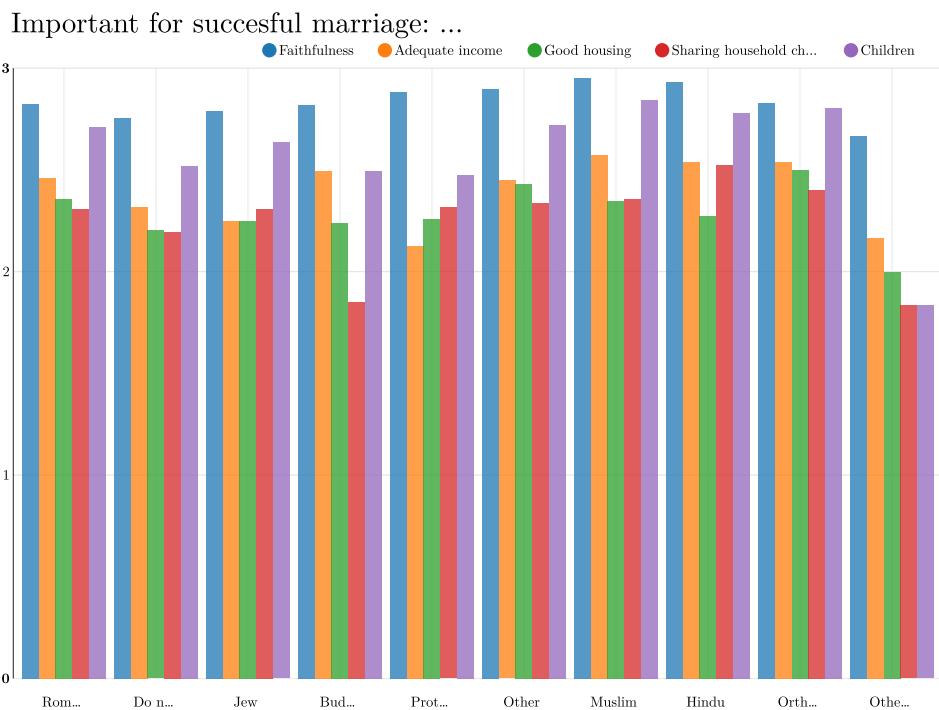
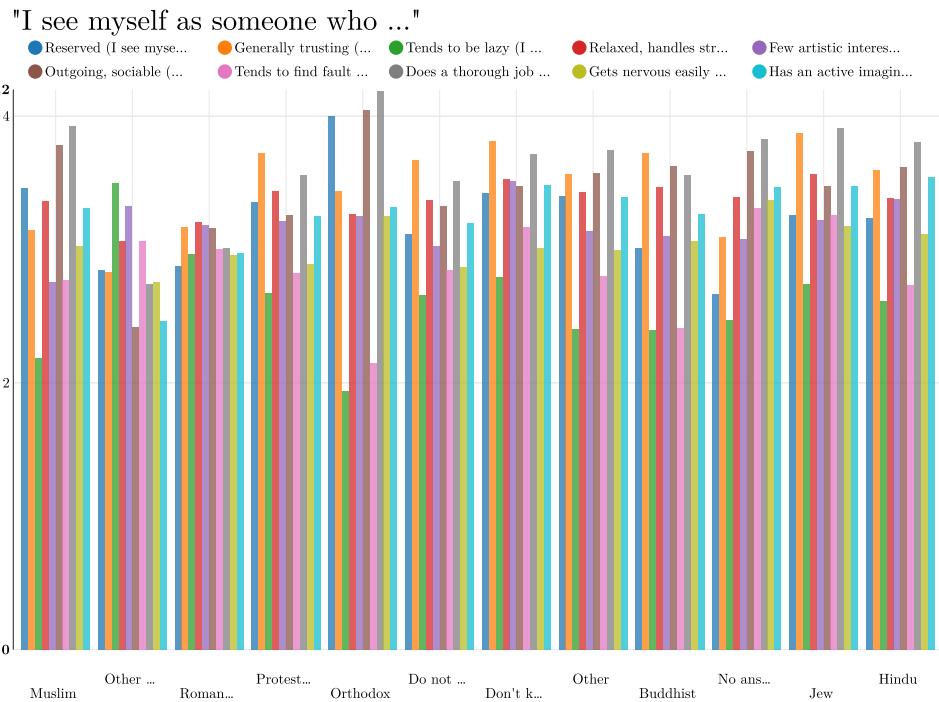
### Most important child qualities

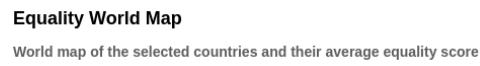
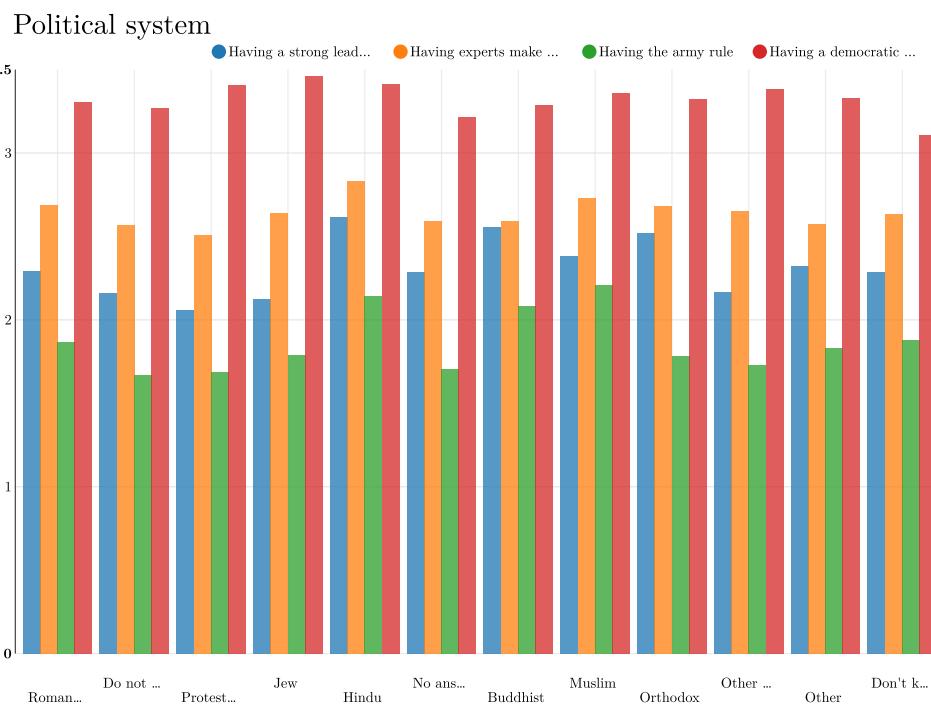


### Justifiable



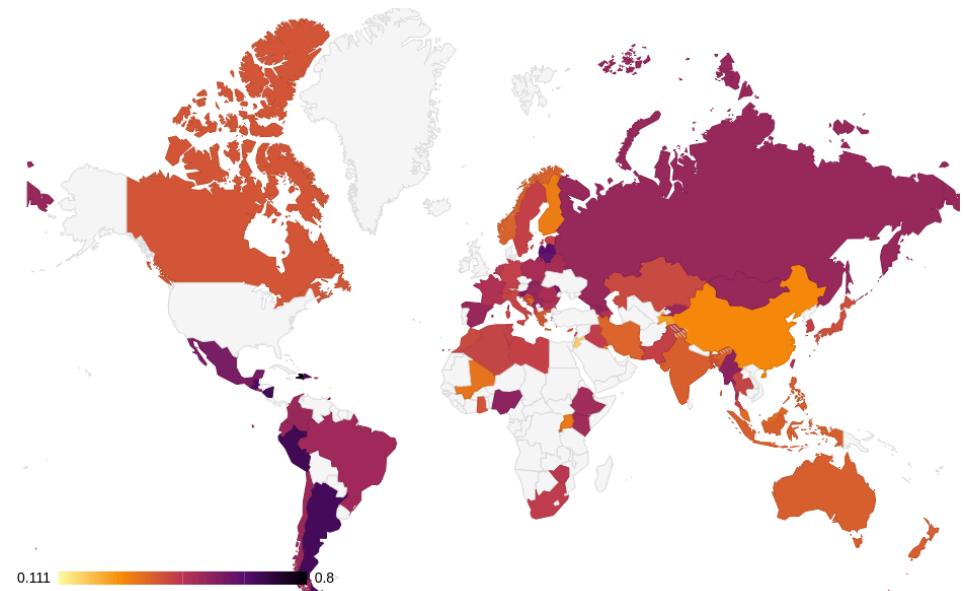




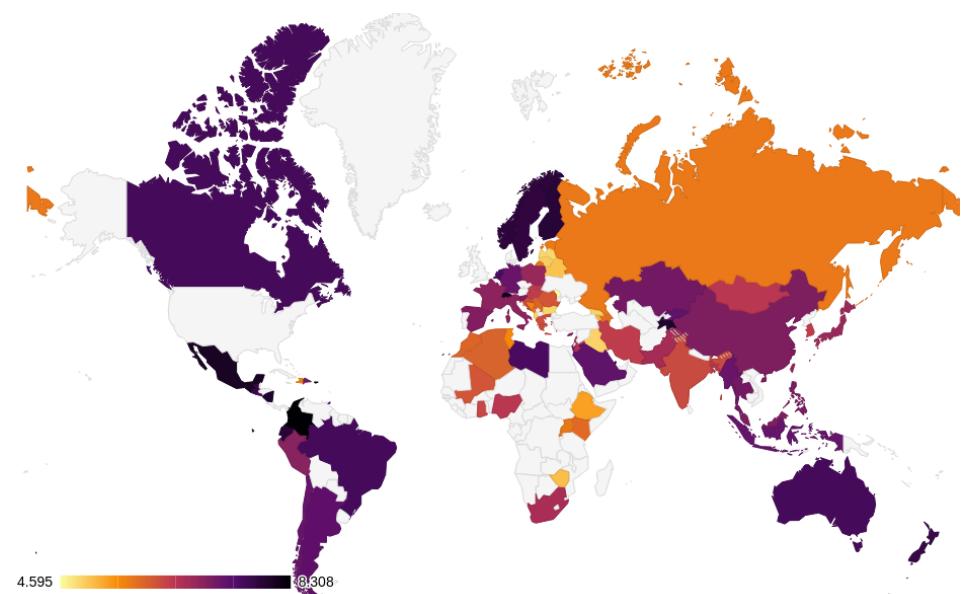


**Scepticism World Map**

World map of the selected countries and their average scepticism score

**Satisfaction World Map**

World map of the selected countries and their average satisfaction with life



## Bibliografia

1. World Values Survey, <https://www.worldvaluessurvey.org/wvs.jsp>
2. Jafarigol, E., Keely, W., Hortag, T., Welborn, T., Hekmatpour, P., Trafalis, T. B.: Religious Affiliation in the Twenty-First Century: A Machine Learning Perspective on the World Value Survey. *Society*. 60, 733–749 (2023). <https://doi.org/10.1007/s12115-023-00887-0>
3. Choropleth Map, [https://hub.knime.com/knime/spaces/Examples/00\\_Components/Visualizations/Choropleth%20Map~L8TzuxeVxB0w4qDm/current-state](https://hub.knime.com/knime/spaces/Examples/00_Components/Visualizations/Choropleth%20Map~L8TzuxeVxB0w4qDm/current-state)
4. Anscombe's quartet, [https://en.wikipedia.org/wiki/Anscombe%27s\\_quartet](https://en.wikipedia.org/wiki/Anscombe%27s_quartet)
5. KNIME: Missing values, <https://www.knime.com/sites/default/files/inline-images/org.knime.base.node.preproc.pmml.missingval.compute.MissingValueHandlerNodeFactory.html>
6. Chawla, N. V., Bowyer, K. W., Hall, L. O., Kegelmeyer, W. P.: SMOTE: Synthetic Minority Over-sampling Technique. *Artificial Intelligence Research*. (2002)
7. Rutecki, M.: SMOTE and Tomek Links for imbalanced data, <https://www.kaggle.com/code/marcinrutecki/smote-and-tomek-links-for-imbalanced-data>
8. KNIME: Creating a Conda environment, [https://docs.knime.com/2019-06/python\\_installation\\_guide/index.html#create\\_env](https://docs.knime.com/2019-06/python_installation_guide/index.html#create_env)
9. Lemaître, G., Nogueira, F., Aridas, C. K.: Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research*. 18, 1–5 (2017)
10. Adekanye, T.: Creating Better Models by Dealing with Class Imbalance in KNIME, <https://medium.com/low-code-for-advanced-data-science/99-accuracy-great-right-18eba5c7564d>
11. Sampling Strategies Comparison, [https://hub.knime.com/victor\\_palacios/spaces/Public/Sampling%20Strategies%20Comparison-ouHL6-Ca44-UK-Wt/current-state](https://hub.knime.com/victor_palacios/spaces/Public/Sampling%20Strategies%20Comparison-ouHL6-Ca44-UK-Wt/current-state)
12. Adekanye, T.: Best Practices with Data Wrangling before running Random Forest Predictions, <https://stats.stackexchange.com/questions/172842/best-practices-with-data-wrangling-before-running-random-forest-predictions>
13. Lichtenhagen, K., Ralf, Hoffmann, G.: HCV data, (2020)
14. Hoffmann, G. F., Bietenbeck, A., Lichtenhagen, R., Klawonn, F.: Using machine learning techniques to generate laboratory diagnostic pathways—a case study. *Journal of Laboratory and Precision Medicine*. (2018)