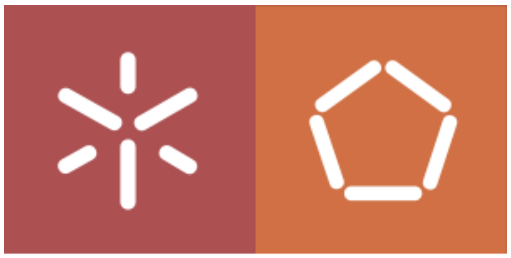


Universidade do Minho
Escola de Engenharia

Mestrado em Engenharia Informática

Perfil de Machine Learning: Fundamentos e Aplicações

Sistemas Baseados em Similaridade

Trabalho Prático de Grupo – 1ª Parte

4º Ano, 1º Semestre

Ano letivo 2020/2021

Diogo Alexandre Rodrigues Lopes

PG42823

Fábio Gonçalves

PG42827

Joel Costa Carvalho

PG42837

Conteúdo

Introdução.

No contexto da Unidade Curricular de Sistemas Baseados em Similaridade, do Perfil de Machine Learning: Fundamentos e Aplicações, foi-nos proposta a realização de um trabalho prático que consistia em desenvolver dois modelos de *machine learning*, utilizando a plataforma *KNIME*, para ambos os *datasets*.

O primeiro modelo de previsão, seria desenvolvido para o *dataset* escolhido pelos docentes desta Unidade Curricular, que visava encontrar a melhor *accuracy* para o número de incidentes rodoviários na cidade de Braga. Ainda neste conjunto de dados, mediante os resultados da *accurary* teríamos de submeter os resultados, de forma personalizada, numa competição online, na plataforma *Kaggle*. O segundo modelo, seria para o *dataset* escolhido por nós, que, da mesma forma que o anterior, visava encontrar a melhor *accuracy* para uma *feature* selecionada por nós.

Para além do objetivo principal, obter a melhor *accuracy,* foi, também, fundamental, saber analisar e interpretar os resultados obtidos nos diversos nodos, em todos os pontos desenvolvidos e, explorar formas de analisar os dados e fazer a sua respetiva limpeza. De salientar que este relatório contém todo o desenvolvimento e resultado obtidos, bem como aspetos importantes relacionados com todo este processo.

Tarefa 1. Análise, Tratamento e Exploração de dados do *dataset* (*2017, 2018 and 2019 Data Professional Salary Results*)

O *dataset* estudado, baseia-se nos resultados da pesquisa salarial de profissionais de dados de 2019, 2018 e 2017 de 84 países. Os dados presentes no dataset contém informações relativas ao:

- Gênero
- Salário
- Atividades Desempenhadas
- Posição na Empresa (Team Leader)
- Principal Base de Dados
- À procura de outro Emprego

Todos os elementos visuais apresentados na resolução desta tarefa estão apresentados num componente.

a. Carregar, no *Knime*, o *dataset* selecionado

Tal como pudemos observar na imagem que se segue, de modo a conseguir ler o *dataset*, utilizamos e implementados o nodo ***Excel Reader (XLS)*** no *workflow*.

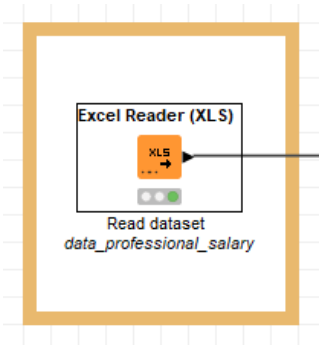


Figura nº1 – Leitura do *dataset*

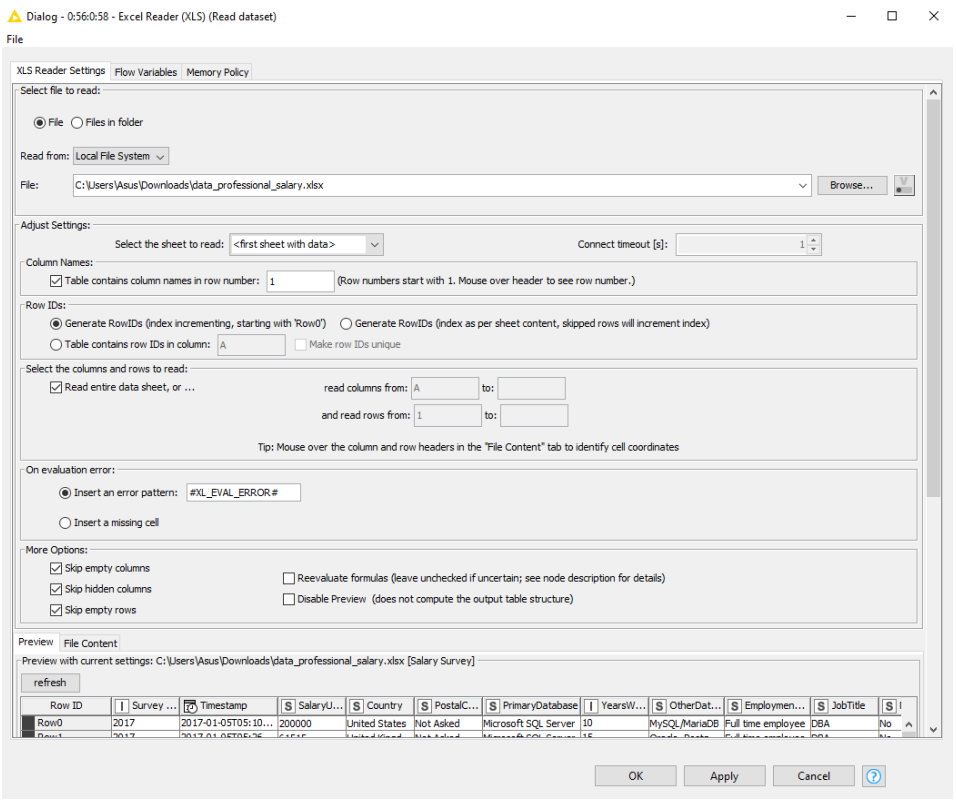
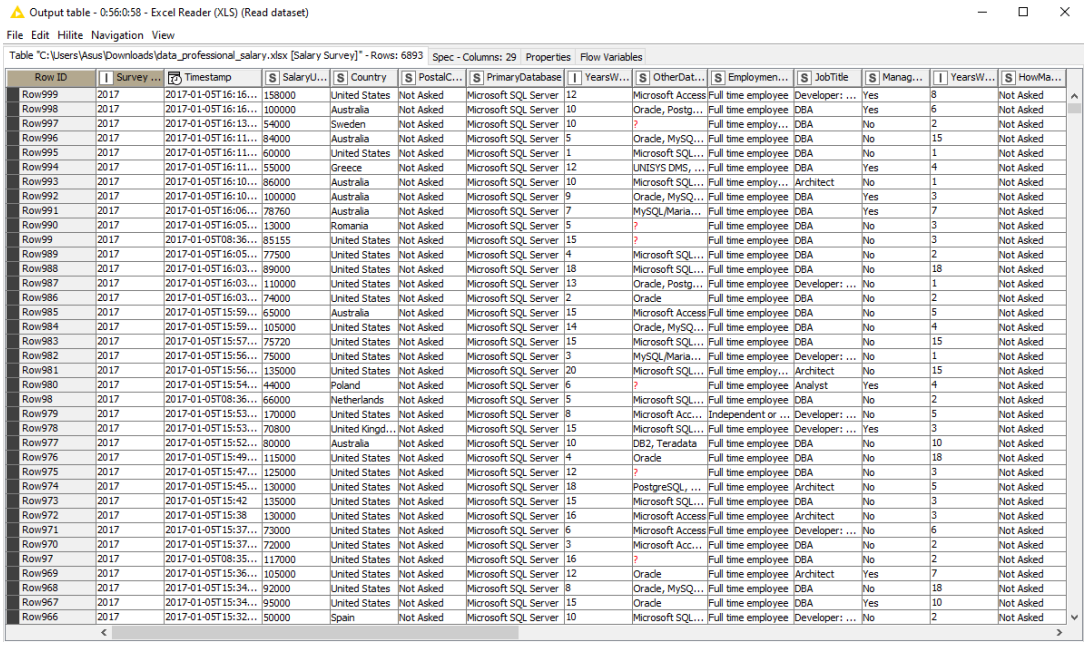


Figura nº2 – Configurações do nodo ***Excel Reader (XLS)***



Row ID	Survey ...	Timestamp	SalaryU...	Country	PostalC...	PrimaryDatabase	YearsW...	OtherDat...	Employmen...	JobTitle	Manag...	YearsW...	HowMa...
Row999	2017	2017-01-05T16:16:...	158000	United States	Not Asked	Microsoft SQL Server	12	Microsoft Access	Full time employee	Developer: ...	Yes	8	Not Asked
Row998	2017	2017-01-05T16:16:...	100000	Australia	Not Asked	Microsoft SQL Server	10	Oracle, Postg...	Full time employee	DBA	Yes	6	Not Asked
Row997	2017	2017-01-05T16:13:...	54000	Sweden	Not Asked	Microsoft SQL Server	10	?	Full time employ...	DBA	No	2	Not Asked
Row996	2017	2017-01-05T16:11:...	84000	Australia	Not Asked	Microsoft SQL Server	5	Oracle, MySQ...	Full time employee	DBA	No	15	Not Asked
Row995	2017	2017-01-05T16:11:...	60000	United States	Not Asked	Microsoft SQL Server	1	Microsoft SQL...	Full time employee	DBA	No	1	Not Asked
Row994	2017	2017-01-05T16:11:...	55000	Greece	Not Asked	Microsoft SQL Server	12	UNISYS DMS, ...	Full time employee	DBA	Yes	4	Not Asked
Row993	2017	2017-01-05T16:10:...	86000	Australia	Not Asked	Microsoft SQL Server	10	Microsoft SQL...	Full time employ...	Architect	No	1	Not Asked
Row992	2017	2017-01-05T16:10:...	100000	Australia	Not Asked	Microsoft SQL Server	9	Oracle, MySQ...	Full time employee	DBA	Yes	3	Not Asked
Row991	2017	2017-01-05T16:06:...	78760	Australia	Not Asked	Microsoft SQL Server	7	MySQL/Maria...	Full time employee	DBA	Yes	7	Not Asked
Row990	2017	2017-01-05T16:05:...	13000	Romania	Not Asked	Microsoft SQL Server	5	?	Full time employee	DBA	No	3	Not Asked
Row99	2017	2017-01-05T08:36:...	85155	United States	Not Asked	Microsoft SQL Server	15	?	Full time employee	DBA	No	3	Not Asked
Row989	2017	2017-01-05T16:05:...	77500	United States	Not Asked	Microsoft SQL Server	4	Microsoft SQL...	Full time employee	DBA	No	2	Not Asked
Row988	2017	2017-01-05T16:03:...	89000	United States	Not Asked	Microsoft SQL Server	18	Microsoft SQL...	Full time employee	DBA	No	18	Not Asked
Row987	2017	2017-01-05T16:03:...	110000	United States	Not Asked	Microsoft SQL Server	13	Oracle, Postg...	Full time employee	Developer: ...	No	1	Not Asked
Row986	2017	2017-01-05T16:03:...	74000	United States	Not Asked	Microsoft SQL Server	2	Oracle	Full time employee	DBA	No	2	Not Asked
Row985	2017	2017-01-05T15:59:...	65000	Australia	Not Asked	Microsoft SQL Server	15	Microsoft Access	Full time employee	DBA	No	5	Not Asked
Row984	2017	2017-01-05T15:59:...	105000	United States	Not Asked	Microsoft SQL Server	14	Oracle, MySQ...	Full time employee	DBA	No	4	Not Asked
Row983	2017	2017-01-05T15:57:...	75720	United States	Not Asked	Microsoft SQL Server	15	Microsoft SQL...	Full time employee	DBA	No	15	Not Asked
Row982	2017	2017-01-05T15:56:...	75000	United States	Not Asked	Microsoft SQL Server	3	MySQL/Maria...	Full time employee	Developer: ...	No	1	Not Asked
Row981	2017	2017-01-05T15:56:...	135000	United States	Not Asked	Microsoft SQL Server	20	Microsoft SQL...	Full time employ...	Architect	No	15	Not Asked
Row980	2017	2017-01-05T15:54:...	44000	Poland	Not Asked	Microsoft SQL Server	6	?	Full time employee	Analyst	Yes	4	Not Asked
Row98	2017	2017-01-05T08:36:...	66000	Netherlands	Not Asked	Microsoft SQL Server	5	Microsoft SQL...	Full time employee	DBA	No	2	Not Asked
Row979	2017	2017-01-05T15:53:...	170000	United States	Not Asked	Microsoft SQL Server	8	Microsoft Acc...	Independent or ...	Developer: ...	No	5	Not Asked
Row978	2017	2017-01-05T15:53:...	70800	United Kingd...	Not Asked	Microsoft SQL Server	15	Microsoft SQL...	Full time employee	Developer: ...	Yes	3	Not Asked
Row977	2017	2017-01-05T15:52:...	80000	Australia	Not Asked	Microsoft SQL Server	10	DB2, Teradata	Full time employee	DBA	No	10	Not Asked
Row976	2017	2017-01-05T15:49:...	115000	United States	Not Asked	Microsoft SQL Server	4	Oracle	Full time employee	DBA	No	18	Not Asked
Row975	2017	2017-01-05T15:47:...	125000	United States	Not Asked	Microsoft SQL Server	12	?	Full time employee	DBA	No	3	Not Asked
Row974	2017	2017-01-05T15:45:...	130000	United States	Not Asked	Microsoft SQL Server	18	PostgreSQL, ...	Full time employee	Architect	No	5	Not Asked
Row973	2017	2017-01-05T15:42	135000	United States	Not Asked	Microsoft SQL Server	15	Microsoft SQL...	Full time employee	DBA	No	3	Not Asked
Row972	2017	2017-01-05T15:38	130000	United States	Not Asked	Microsoft SQL Server	16	Microsoft Access	Full time employee	Architect	No	3	Not Asked
Row971	2017	2017-01-05T15:37:...	73000	United States	Not Asked	Microsoft SQL Server	6	Microsoft Access	Full time employee	Developer: ...	No	6	Not Asked
Row970	2017	2017-01-05T15:37:...	72000	United States	Not Asked	Microsoft SQL Server	3	Microsoft Acc...	Full time employee	DBA	No	2	Not Asked
Row97	2017	2017-01-05T08:35:...	117000	United States	Not Asked	Microsoft SQL Server	16	?	Full time employee	DBA	No	2	Not Asked
Row969	2017	2017-01-05T15:36:...	105000	United States	Not Asked	Microsoft SQL Server	12	Oracle	Full time employee	Architect	Yes	7	Not Asked
Row968	2017	2017-01-05T15:34:...	92000	United States	Not Asked	Microsoft SQL Server	8	Oracle, MySQ...	Full time employee	DBA	No	18	Not Asked
Row967	2017	2017-01-05T15:34:...	95000	United States	Not Asked	Microsoft SQL Server	15	Oracle	Full time employee	DBA	Yes	10	Not Asked
Row966	2017	2017-01-05T15:32:...	50000	Spain	Not Asked	Microsoft SQL Server	10	Microsoft SQL...	Full time employee	Developer: ...	No	2	Not Asked

Figura nº3 – Output Table do *dataset*

b. Aplicar nodos, de modo fazer o Tratamento dos Dados

Com o objetivo de fazer o tratamento dos dados presentes no *dataset*, foram aplicados vários nodos, como por exemplo, nodos para tratamento de ***Missing Values***, nodos para ***extração de datas***, nodos para ***substituição de Strings***, entre outros. Todos estes nodos implementados para tratamento de dados, encontram-se devidamente evidenciados e explicados nos pontos que se seguem. De salientar que os já referidos nodos, encontram-se dentro de um ***metanode*** criado (***Metanode work data***).

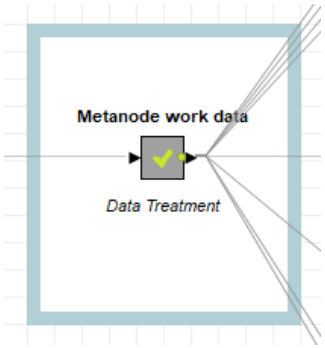


Figura nº4 – ***Metanode*** Implementado para Tratamento de Dados

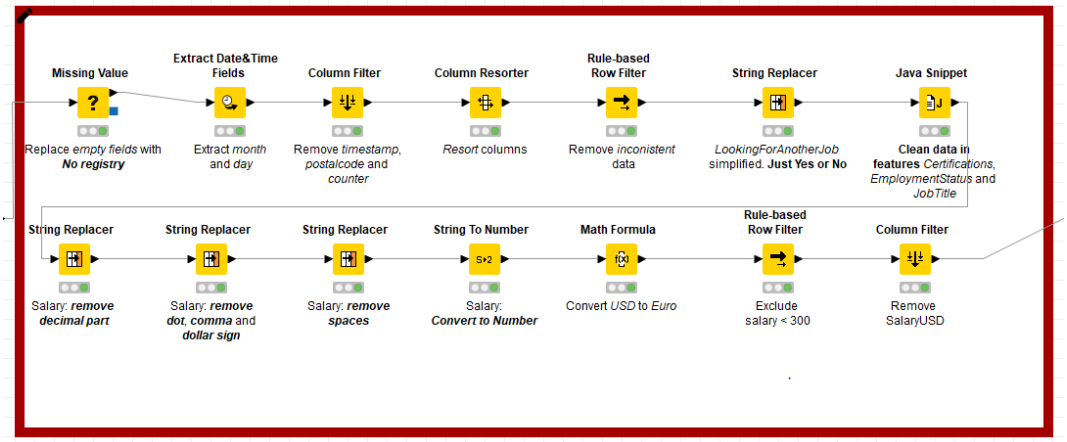


Figura nº5 –*Workfkow* Implementado para Tratamento de Dados

1. Tratar os valores em falta

No *dataset* selecionado, existem registos onde há campos sem quaisquer tipo de valores atribuídos, o que nos obriga a fazer tratamento dos mesmos. Deste modo, foi implementado no *workflow* um nodo ***Missing Value***, cujo objetivo é substituir todos os campos vazios pela String '***No registry***'.

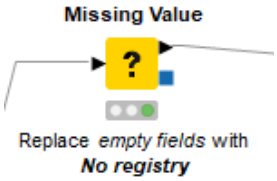


Figura nº6 – Tratamento dos Valores em Falta

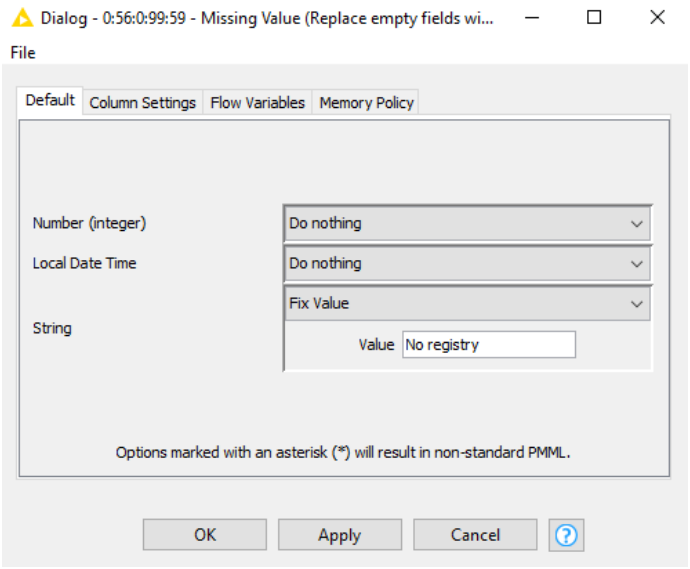


Figura nº7 – Configurações do nodo ***Missing Value***

2. Extração do dia e do mês do atributo data

Uma vez que as datas presentes nos registos do *dataset* encontram-se no formato *Timestamp*, utilizando o nodo ***Extract Date&Time Fields***, conseguimos obter/extrair o dia do mês e o mês(número), que posteriormente será utilizado na análise dados.

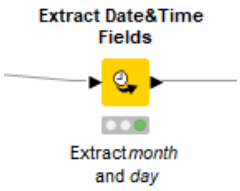


Figura nº8 – Extração do Dia e Mês do Campo Data

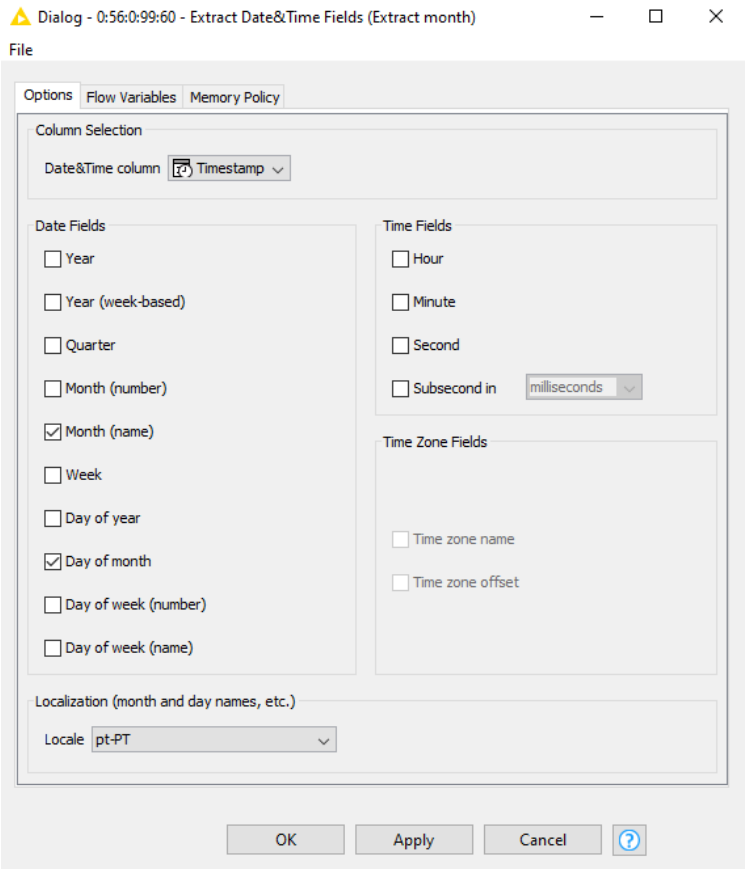


Figura nº9 – Configurações do nodo ***Extract Date&Time Fields***

3. Remoção de atributos (*timestamp*, *postalcode* e *counter*)

Como não pretendemos ter atributos que não acrescentem valor ao nosso modelo, implementamos o nodo ***Column Filter*** ao nosso *workflow* para remover atributos do mesmo. As *features* que removemos foram o *Timestamp*, *Postalcode* e *Counter*.

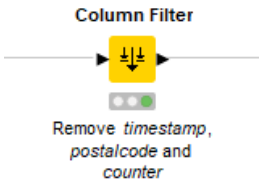


Figura nº10 – Remoção de Atributos (*timestamp*, *postalcode* e *counter*)

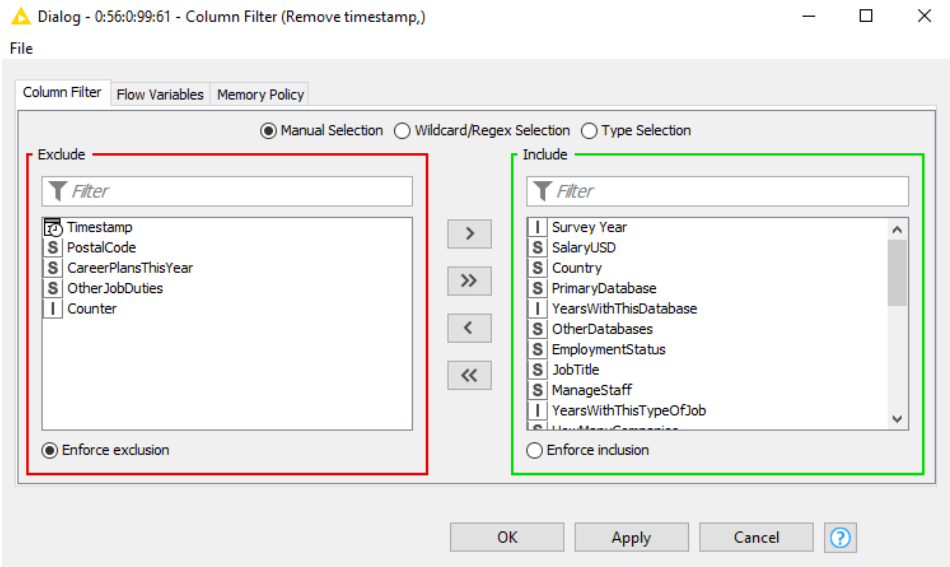


Figura nº11 – Configurações do nodo ***Column Filter***

4. Remoção de dados inconsistentes

Recorrendo ao nodo ***Rule-based Row Filter***, removemos os dados que na nossa opinião são inconsistentes no nosso modelo, uma vez que na nossa opinião dificilmente uma pessoa trabalha mais de 60 horas por semana e também não acreditamos que existam pessoas com 45 anos de experiência sempre com a mesma base de dados.

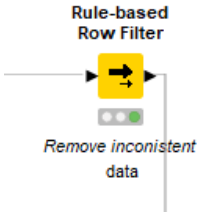


Figura nº12 – Remoção de Dados Inconsistentes

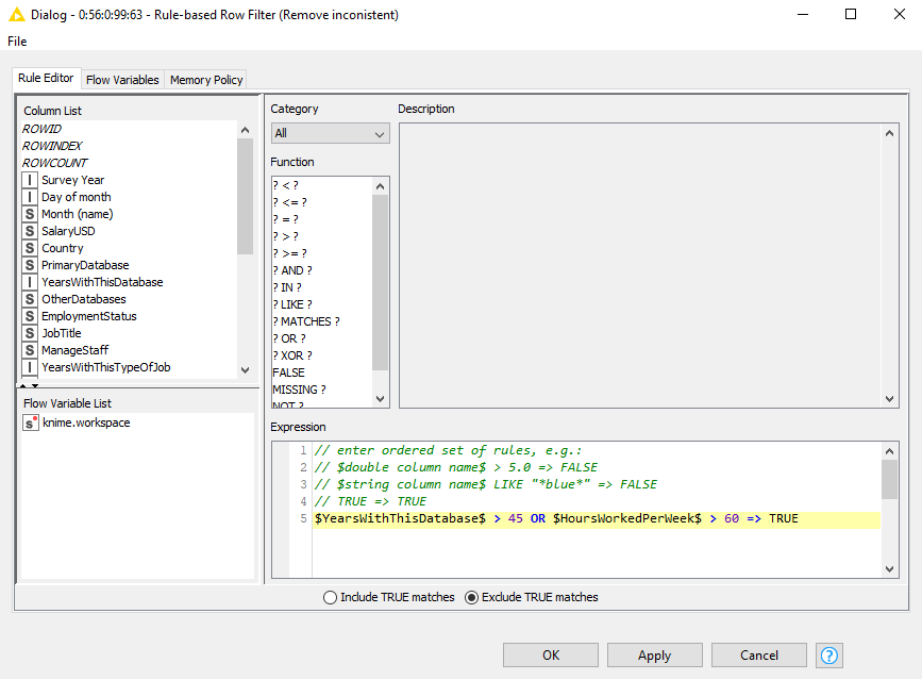


Figura nº13 – Configurações do nodo ***Rule-based Row Filter***

5. Simplificação da feature '*LookingForAnotherJob*'

Com a implementação do nodo ***String Replacer*** no nosso *workflow*, conseguimos simplificar o atributo '*LookingForAnotherJob*', uma vez que se tratava de um atributo bastante complexo.

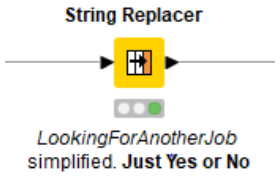


Figura nº14 – Simplificação da feature '*LookingForAnotherJob*'

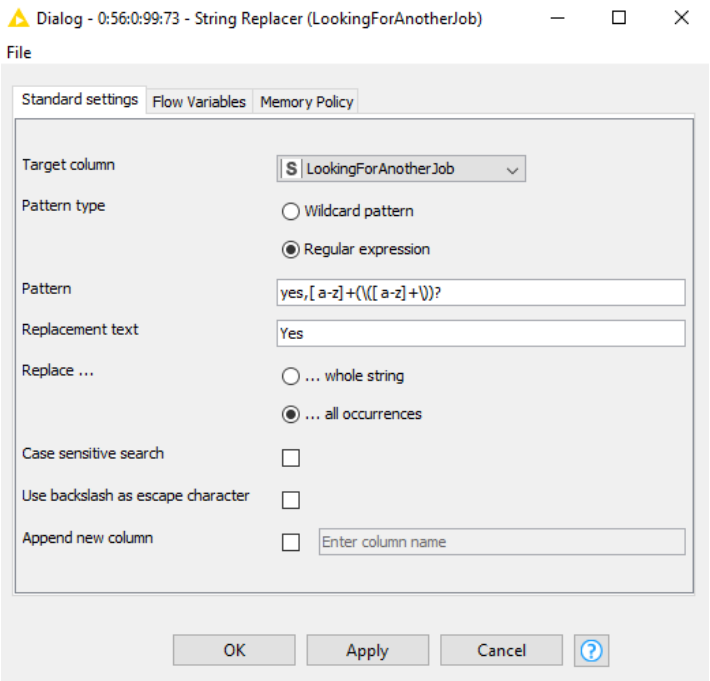


Figura nº15 – Configurações do nodo ***String Replacer***

6. Limpar dados das *features*

Com o objetivo de fazer a limpeza das features do dataset, recorremos ao nodo ***Java Snippet***, tendo em consideração as configurações apresentadas nas imagens seguintes. É importante salientar que *feature* ***Certifications*** apenas passou a ter valores de *'yes'* e *'no'*, a *feature* ***EmploymentStatus*** apenas passou a ter valores de *'part-time'* e *'full-time'* e na *feature* ***JobTitle*** foi feita simplificação de 6 tipos de profissionais, uma vez que continha informação redundante.

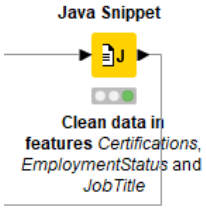


Figura nº16 – Nodo Aplicado para Limpeza de Dados do *dataset*

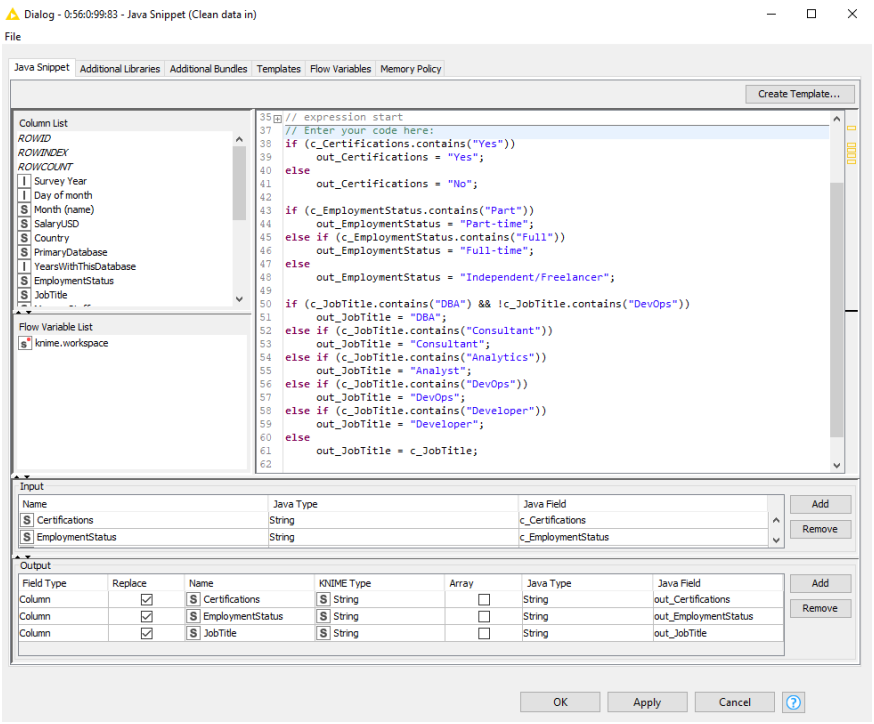


Figura nº17 – Configurações do nodo ***Java Snippet***

7. Tratamento do atributo salário

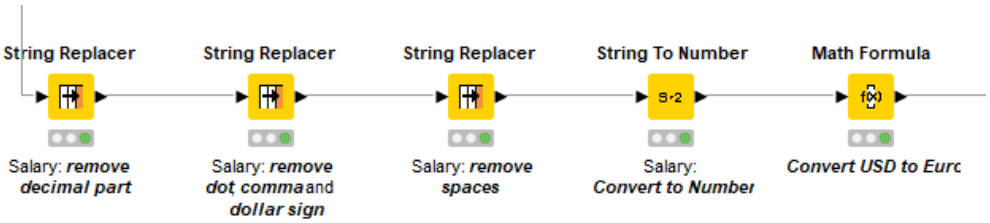


Figura nº18 – Tratamento do Atributo Salário

I. Remoção da parte decimal do salário

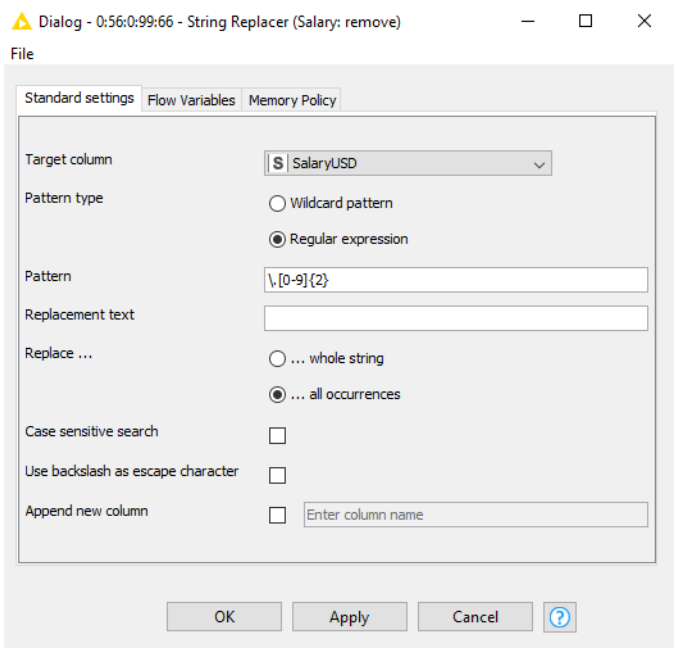


Figura nº19 – Configurações do nodo ***String Replacer***

II. Remoção do '.', da ',' e do '$' do salário

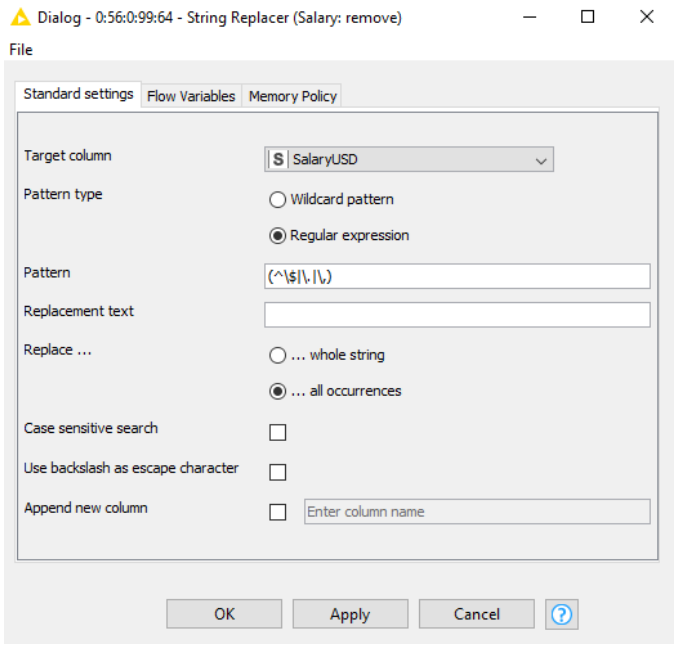


Figura nº20 – Configurações do nodo ***String Replacer***

III. Remoção de ‘ ‘ (espaços) do salário

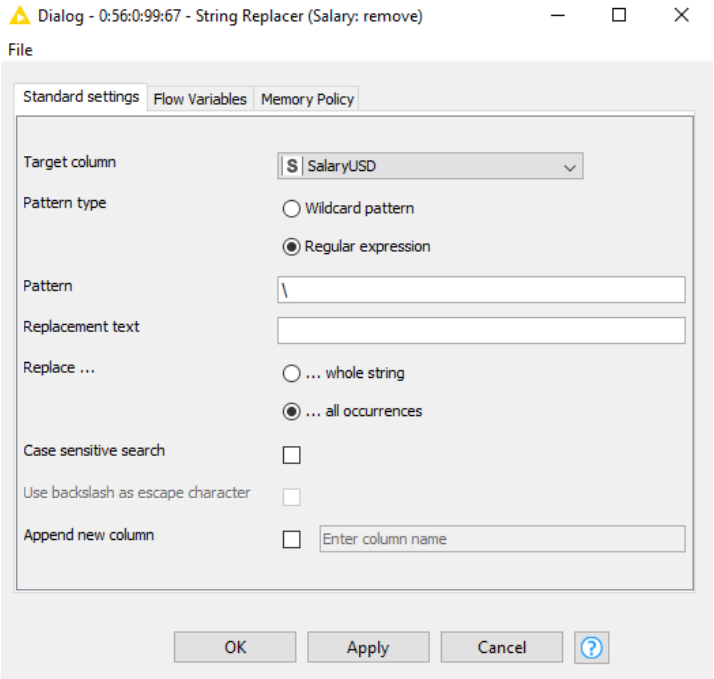


Figura nº21 – Configurações do nodo ***String Replacer***

IV. Converter o salário para um *number*

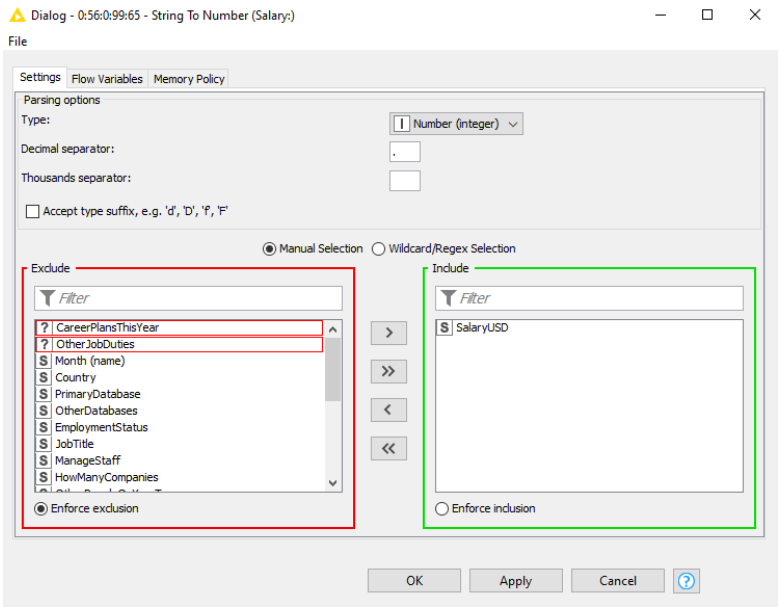


Figura nº22 – Configurações do nodo ***String To Numberr***

V. Converter o *USD* para *Euro*

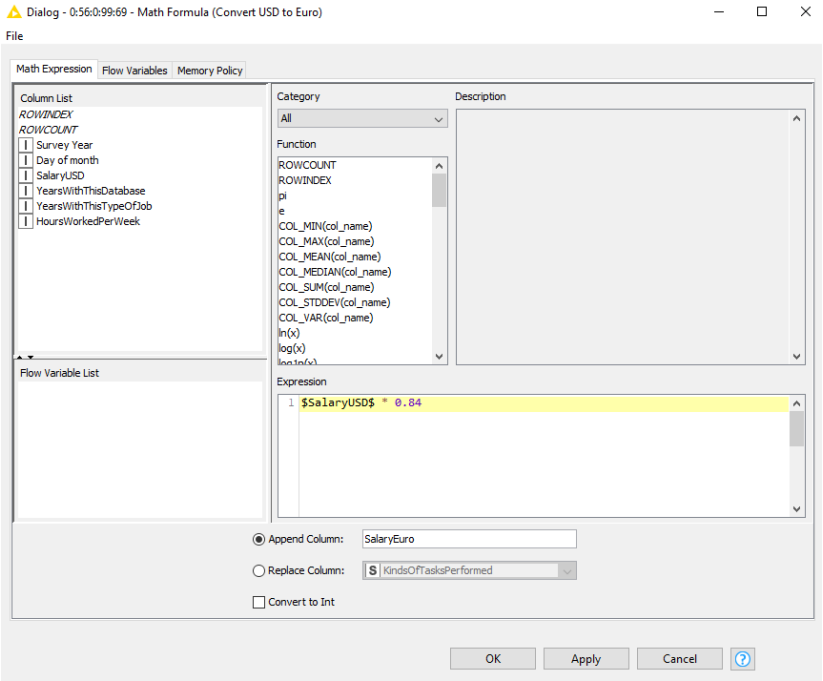


Figura nº23 – Configurações do nodo ***Math Formula***

8. Exclusão de salários inferiores a 300

Como não achamos relevante para a implementação do nosso modelo de previsão incluir os salários inferiores a 300, recorrendo ao nodo ***Rule-based Row Filter*** para fazer a remoção dos mesmos.

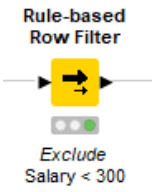


Figura nº24 – Exclusão de Salários Inferiores a 300

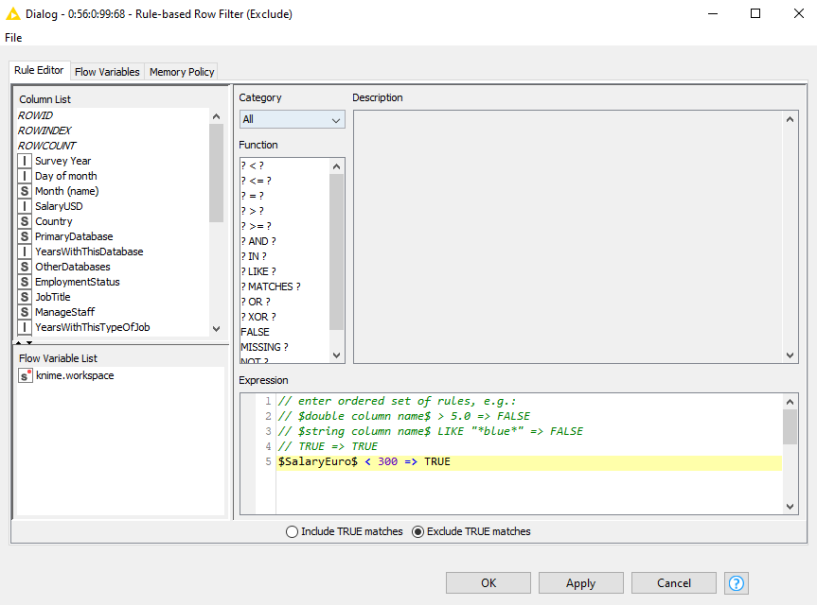


Figura nº25 – Configurações do nodo ***Rule-based Row Filter***

9. Remoção de salários do tipo '*SalaryUSD*'

À semelhança do que foi explicado no tópico anterior, também não achamos relevante para a implementação do nosso modelo de previsão incluir os salários do tipo *USD*, e para isso recorremos ao nodo ***Column Filter*** para fazer a remoção dos mesmos.

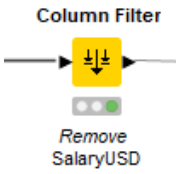


Figura nº26 – Remoção de Salários do tipo '*SalaryUSD*'

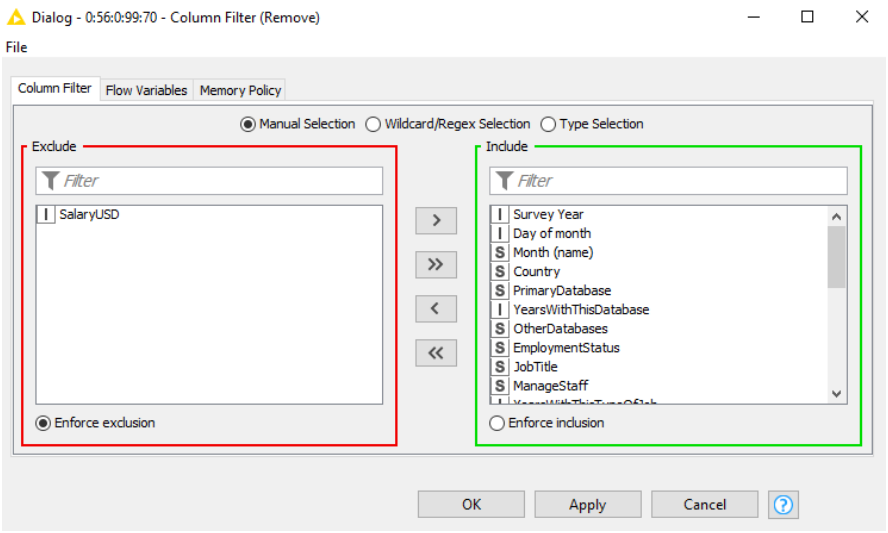


Figura nº27 – Configurações do nodo ***Column Filter***

c. Aplicar nodos, de modo fazer a Análise de *Features* do Modelo

Com o objetivo de fazer a análise de dados, foram aplicados vários nodos, de modo a conseguir perceber quais seriam as *features* ideais para o nosso modelo. Todos estes nodos implementados no *workflow*, encontram-se dentro de um ***metanode*** criado (***Metanode Analyze Features***), e estão devidamente explicados neste ponto relatório prático.

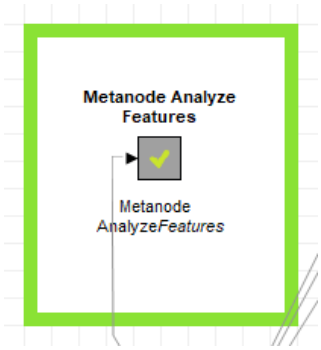


Figura nº28 – ***Metanode*** Implementado para Análise de Dados

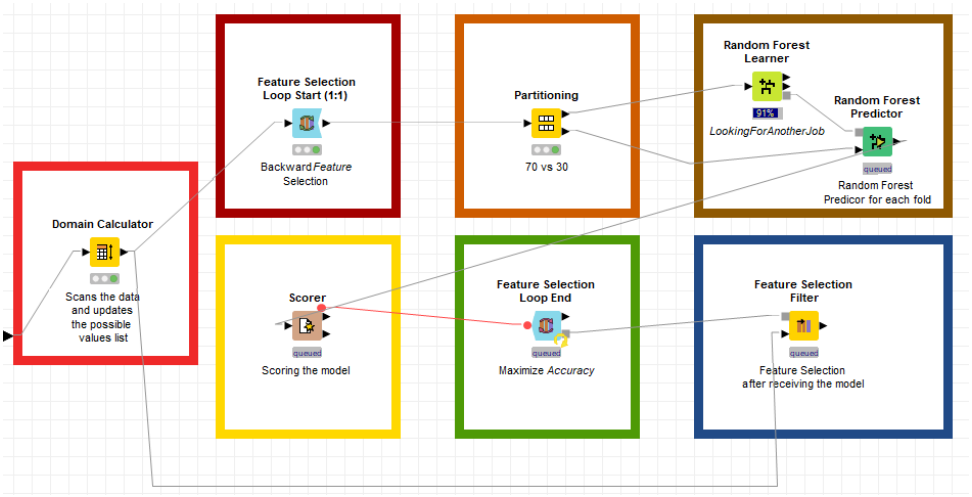


Figura nº29 –*Workfkow* Implementado para Análise de Dados

De modo a conseguirmos fazer a *backward feature selection*, implementamos no *workflow* o nodo ***Feature Selection Loop Start***. Na configuração deste nodo apenas excluímos o atributo '*LookingForAnotherJob*'.

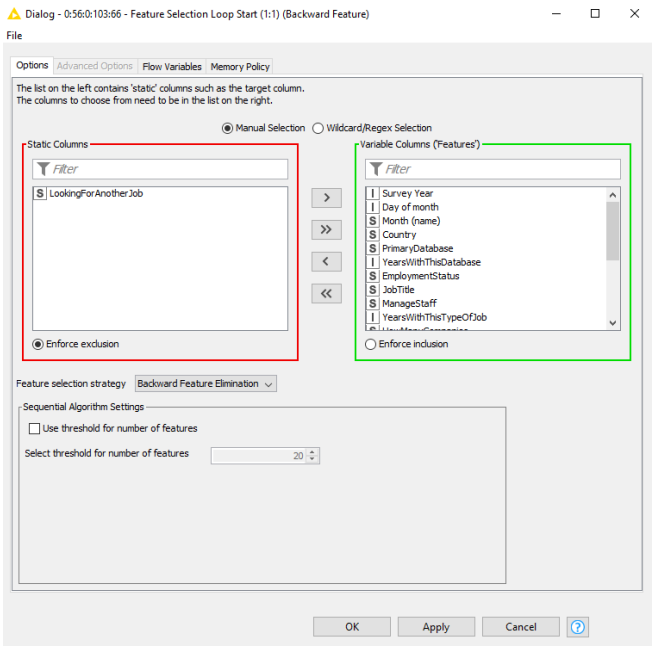


Figura nº30 – Configurações do nodo ***Feature Selection Loop Start***

Recorrendo ao nodo ***Partitioning***, foi feito o particionamento do conjunto de dados, de acordo com as configurações apresentadas na imagem seguinte.

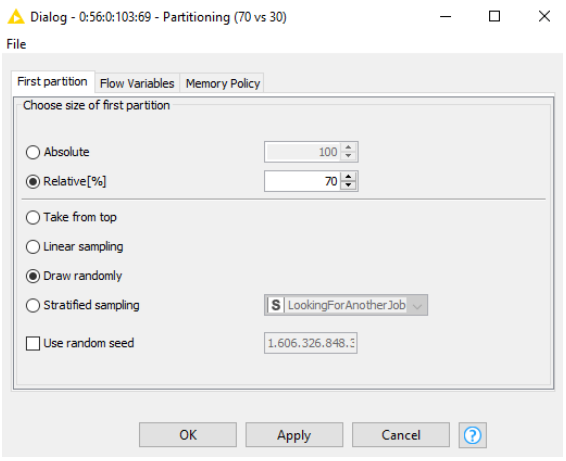


Figura nº31 – Configurações do nodo ***Partitioning***

No que diz respeito à configuração do nodo ***Random Forest Learner*** implementado, é importante salientar que a *Split Criterion* selecionada foi a ***Gini index***, ***com 180 modelos***. As restantes configurações efetuadas no nodo podem ser observadas na imagem seguinte.

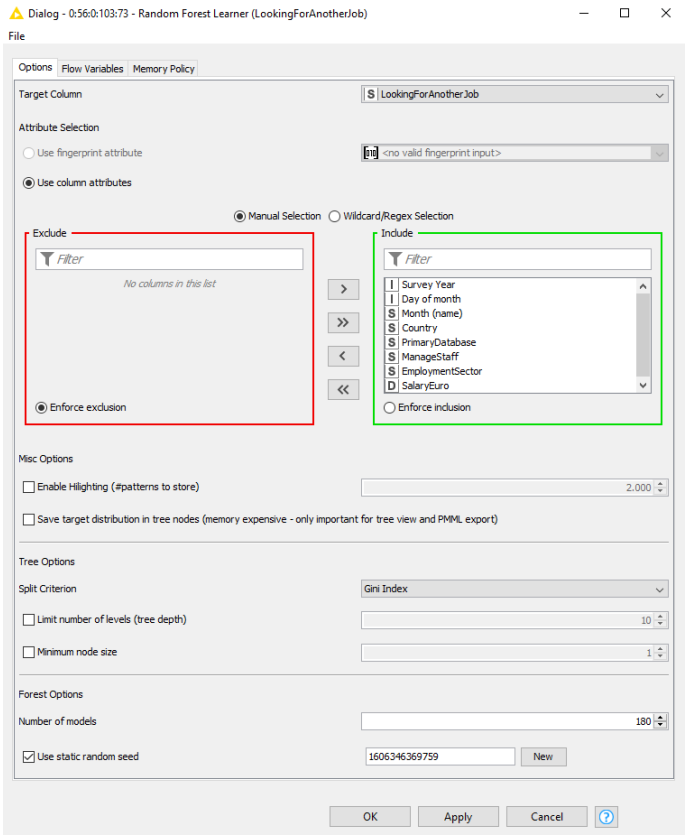


Figura nº32 – Configurações do nodo ***Random Forest Learner***

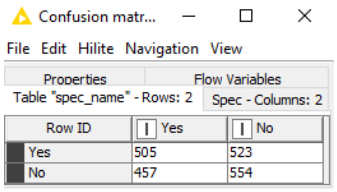


Figura nº33 – *Confusion Matrix* Obtida no nodo ***Scorer***

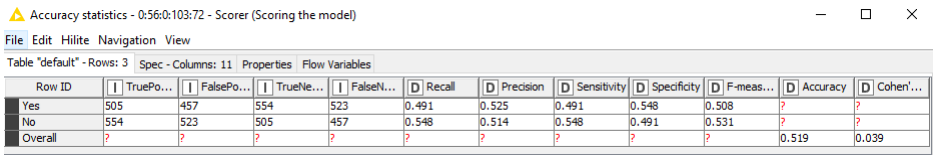


Figura nº34 – *Accuracy Statistics* Obtidas no nodo ***Scorer***

Por fim, recorrendo ao nodo ***Feature Selection Filter*** implementado no *workflow*, foi feita a seleção de features depois de receber o modelo.

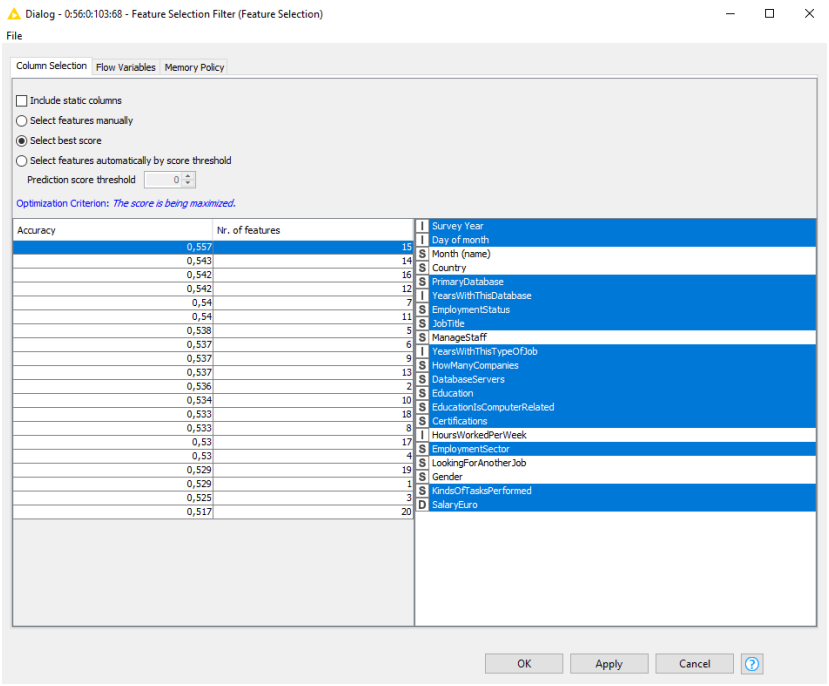
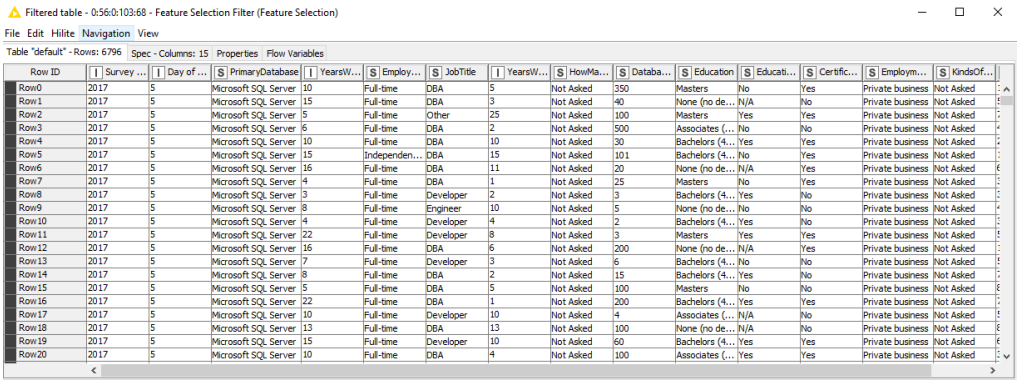


Figura nº35 – Configurações do nodo ***Feature Selection Filter***



Row ID	Survey ...	Day of ...	PrimaryDatabase	YearsW...	Employ...	JobTitle	YearsW...	HowMa...	Databa...	Education	Educati...	Certific...	Employm...	KindsOf...
Row0	2017	5	Microsoft SQL Server	10	Full-time	DBA	5	Not Asked	350	Masters	No	Yes	Private business	Not Asked
Row1	2017	5	Microsoft SQL Server	15	Full-time	DBA	3	Not Asked	40	None (no de...	N/A	No	Private business	Not Asked
Row2	2017	5	Microsoft SQL Server	5	Full-time	Other	25	Not Asked	100	Masters	Yes	Yes	Private business	Not Asked
Row3	2017	5	Microsoft SQL Server	6	Full-time	DBA	2	Not Asked	500	Associates (...	No	No	Private business	Not Asked
Row4	2017	5	Microsoft SQL Server	10	Full-time	DBA	10	Not Asked	30	Bachelors (4...	Yes	Yes	Private business	Not Asked
Row5	2017	5	Microsoft SQL Server	15	Independen...	DBA	15	Not Asked	101	Bachelors (4...	No	Yes	Private business	Not Asked
Row6	2017	5	Microsoft SQL Server	16	Full-time	DBA	11	Not Asked	20	None (no de...	N/A	Yes	Private business	Not Asked
Row7	2017	5	Microsoft SQL Server	4	Full-time	DBA	1	Not Asked	25	Masters	No	Yes	Private business	Not Asked
Row8	2017	5	Microsoft SQL Server	3	Full-time	Developer	2	Not Asked	3	Bachelors (4...	Yes	No	Private business	Not Asked
Row9	2017	5	Microsoft SQL Server	8	Full-time	Engineer	10	Not Asked	5	None (no de...	No	No	Private business	Not Asked
Row10	2017	5	Microsoft SQL Server	4	Full-time	Developer	4	Not Asked	2	Bachelors (4...	Yes	No	Private business	Not Asked
Row11	2017	5	Microsoft SQL Server	22	Full-time	Developer	8	Not Asked	3	Masters	Yes	Yes	Private business	Not Asked
Row12	2017	5	Microsoft SQL Server	16	Full-time	DBA	6	Not Asked	200	None (no de...	N/A	Yes	Private business	Not Asked
Row13	2017	5	Microsoft SQL Server	7	Full-time	Developer	3	Not Asked	6	Bachelors (4...	No	No	Private business	Not Asked
Row14	2017	5	Microsoft SQL Server	8	Full-time	DBA	2	Not Asked	15	Bachelors (4...	Yes	No	Private business	Not Asked
Row15	2017	5	Microsoft SQL Server	5	Full-time	DBA	5	Not Asked	100	Masters	No	No	Private business	Not Asked
Row16	2017	5	Microsoft SQL Server	22	Full-time	DBA	1	Not Asked	200	Bachelors (4...	Yes	Yes	Private business	Not Asked
Row17	2017	5	Microsoft SQL Server	10	Full-time	Developer	10	Not Asked	4	Associates (...	N/A	No	Private business	Not Asked
Row18	2017	5	Microsoft SQL Server	13	Full-time	DBA	13	Not Asked	100	None (no de...	N/A	No	Private business	Not Asked
Row19	2017	5	Microsoft SQL Server	15	Full-time	Developer	10	Not Asked	60	Bachelors (4...	Yes	Yes	Private business	Not Asked
Row20	2017	5	Microsoft SQL Server	10	Full-time	DBA	4	Not Asked	100	Associates (...	Yes	Yes	Private business	Not Asked

Figura nº36 – *Result Table* Obtida no nodo ***Feature Selection Filter***

d. Aplicar nodos, de modo fazer o *Tuning* do Modelo

Com o objetivo de fazer o *tuning* aos dados, foram aplicados vários nodos, de modo a conseguir perceber quais seriam as configurações ideais para o nosso modelo. Todos estes nodos implementados no *workflow*, encontram-se dentro de um ***metanode*** criado (***Metanode Tuning***), e estão devidamente explicados neste ponto relatório prático.

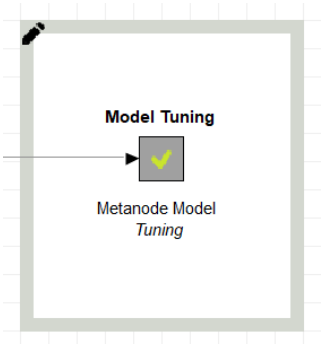


Figura nº37 – ***Metanode*** Implementado para Tuning de Dados

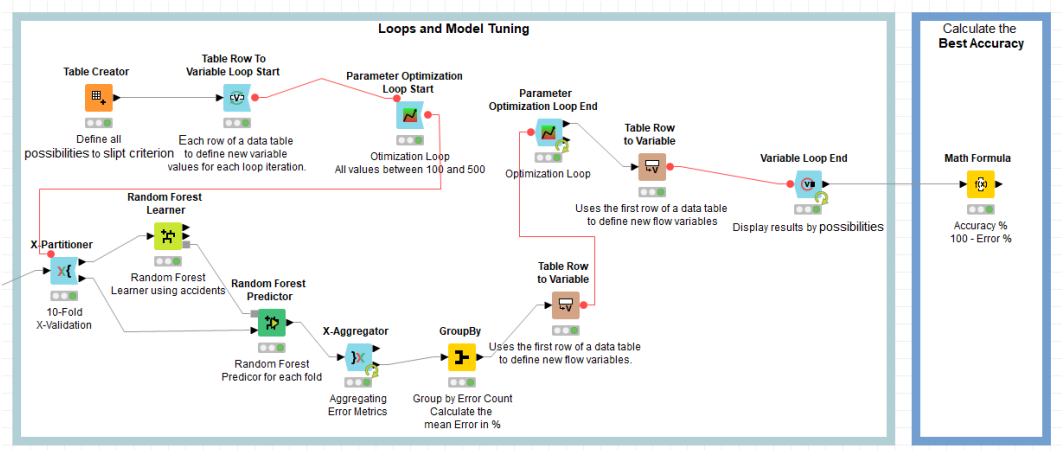


Figura nº38 –*Workfkow* Implementado para Tuning de Dados

Na imagem seguinte é possível observar as configurações feitas no nodo ***Table Creator***, sendo que o objetivo do mesmo é definir todas as possibilidades para o *Split Criterion*.

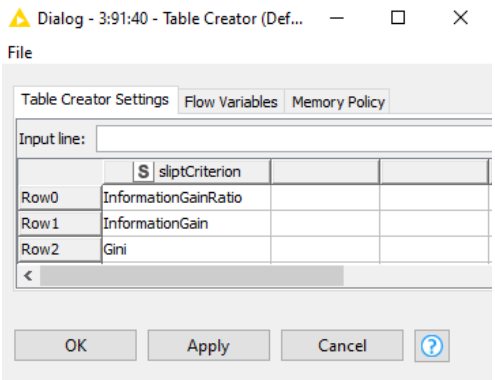


Figura nº39 – Configurações do nodo ***Table Creator***

O nodo ***Table Row To Variable Loop Start*** foi implementado no *workflow* com o objetivo de que em cada linha de uma tabela, fosse definida uma nova variável com valores para cada iteração do *loop*.

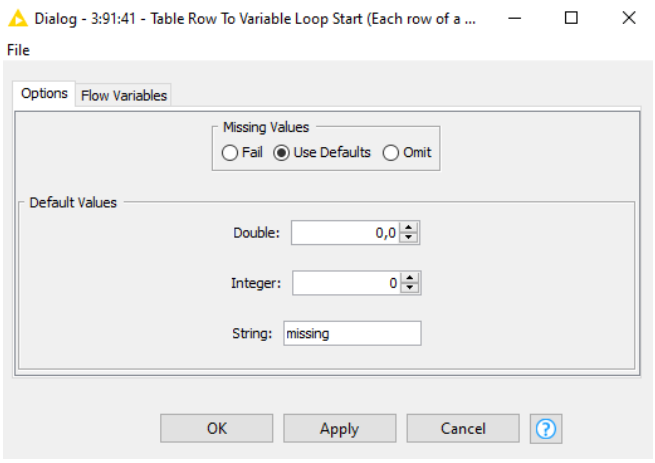


Figura nº40 – Configurações do nodo ***Table Row To Variable Loop Start***

De acordo com as configurações referentes ao nodo ***Parameter Optimization Loop Start***, conseguimos perceber que este nodo foi implementado no *workflow* com o objetivo de fazer a otimização do *loop* para todos os valores entre 100 e 200.

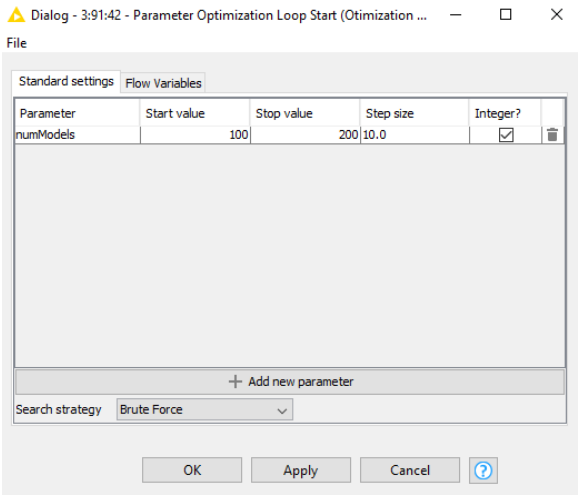


Figura nº41 – Configurações do nodo ***Parameter Optimization Loop Start***

Para fazer uma *X-Validation*, implementamos o nodo ***X-Partitioner*** no nosso *workflow*. Para fazer a agregação, implementamos o ***X-Aggregator***.

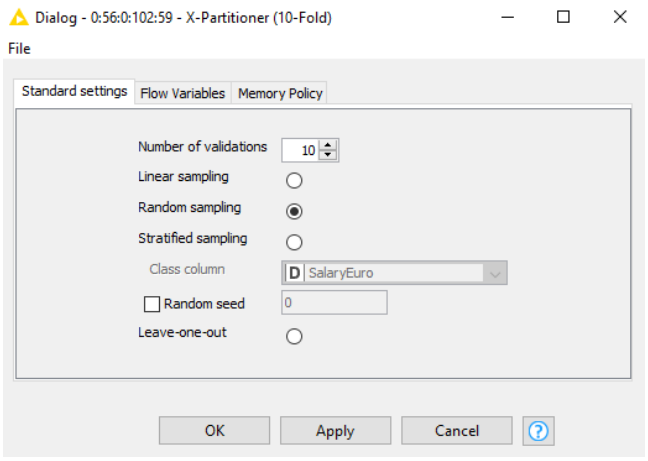


Figura nº42 – Configurações do nodo ***X-Partitioner***

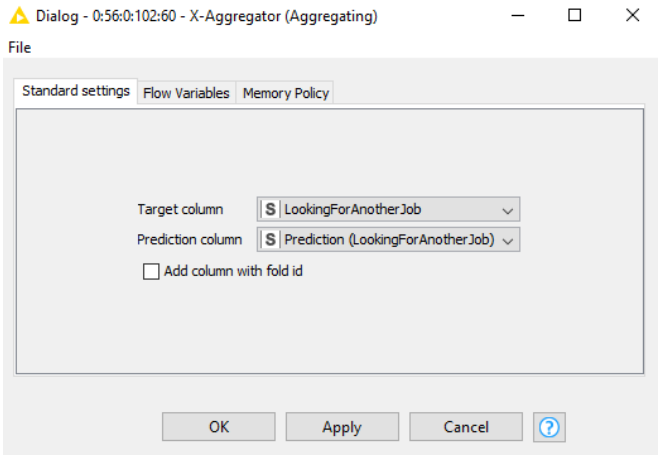


Figura nº43 – Configurações do nodo ***X-Aggregator***

Uma vez que decidimos agrupar por contagem de erros e calcular o erro médio em %, implementamos no nosso *workflow* o nodo ***Group By***.

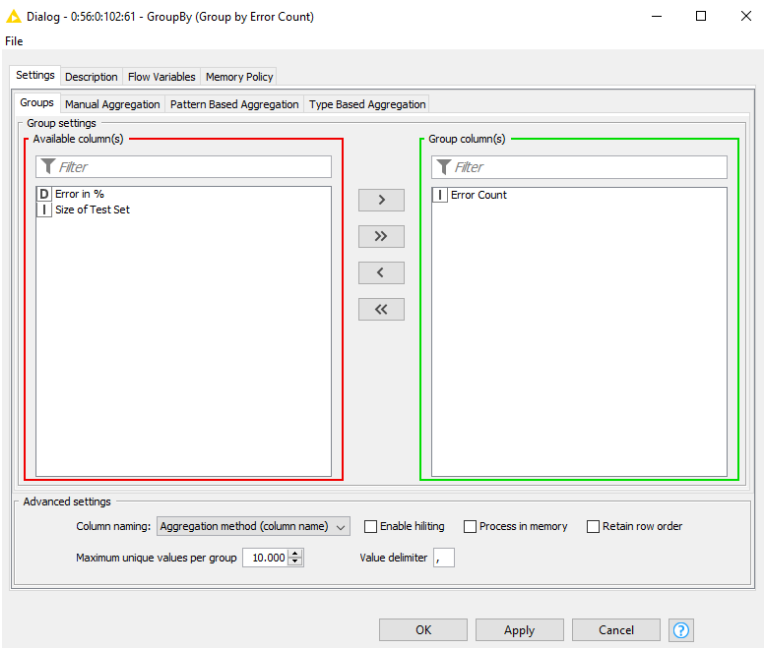


Figura nº44 – Configurações do nodo ***Table Row to Variable***

O nodo ***Table Row to Variable*** foi implementado com o objetivo de utilizar os primeiros dados da tabela, de modo a definir novas *flow variables*.

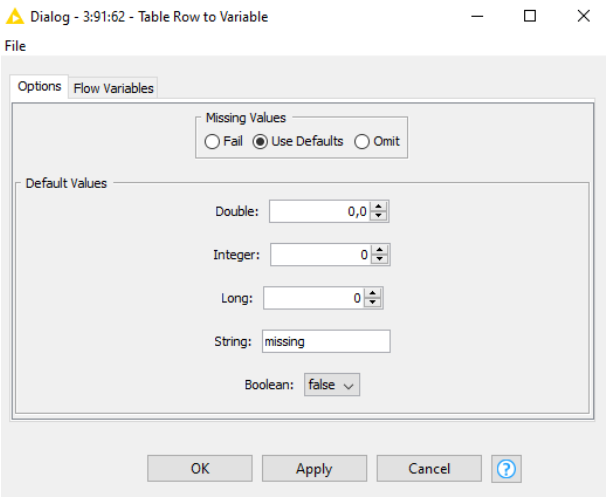


Figura nº45 – Configurações do nodo ***Table Row to Variable***

De modo a otimizar os *loops*, foi implementado o nodo ***Parameter Optimization Loop End*** no *workflow*, sendo que a *flow variable* na função objetivo era a média do erro, sendo que era pretendido a minimização da mesma.

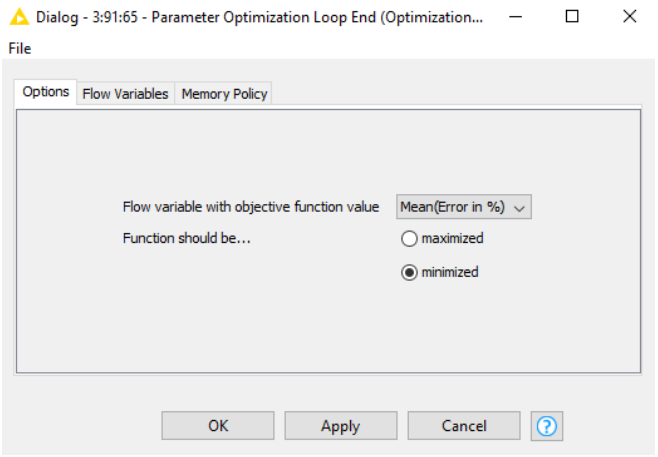


Figura nº46 – Configurações do nodo ***Parameter Optimization Loop End***

À semelhança do outro nodo ***Table Row to Variable***, este foi igualmente implementado com o objetivo de utilizar os primeiros dados da tabela, de modo a definir novas *flow variables*.

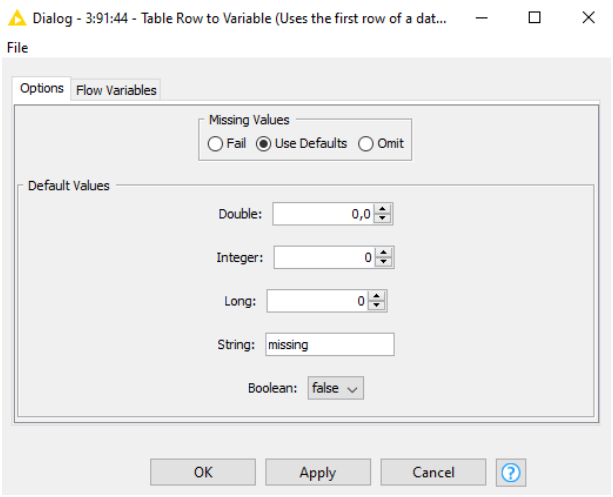


Figura nº47 – Configurações do nodo ***Table Row to Variable***

Este nodo (***Variable Loop End***) foi implementado com o intuito de fazer o display dos resultados de todas as opções possíveis para o modelo a implementar.

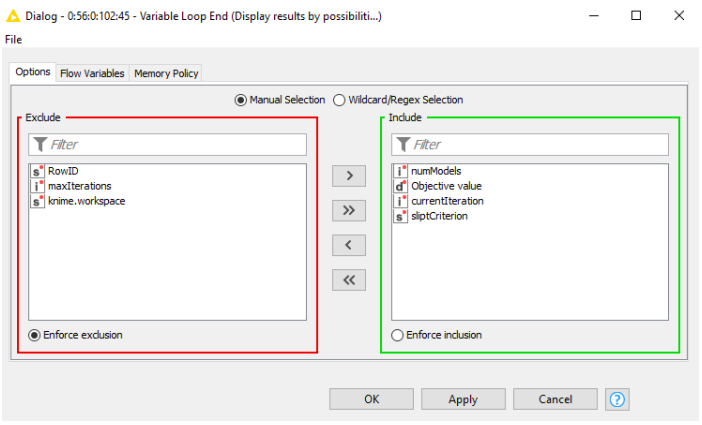


Figura nº48 – Configurações do nodo ***Variable Loop End***

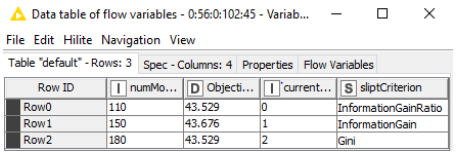


Figura nº49 – *Output* do Gerado no nodo ***Variable Loop End***

De acordo com as configurações apresentadas na imagem seguinte, é possível observar que o nodo ***Math Formula*** foi implementado com o objetivo de conseguirmos perceber qual é a nossa *accuracy* numa escala de 0% a 100%.

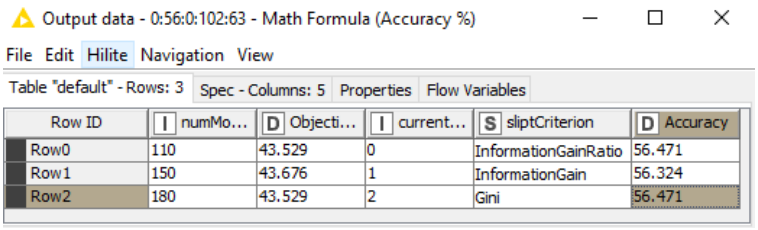


Figura nº50 – Configurações do nodo ***Math Formula***

Conforme a figura acima, a melhor *accuracy* foi obtida com o *Split Criterion* ***Gini*** com o **número** de modelos a **180**.

e. Aplicar nodos, de modo fazer a Exploração de Dados

1. Qual o país que paga melhor?

Foram implementados os nodos ***GropuBy***, ***Bar Chart*** e ***Column Filter*** de modo a conseguirmos responder a esta questão. Após análise dos resultados obtido pudemos observar que o país que paga melhor são os Hong Kong.

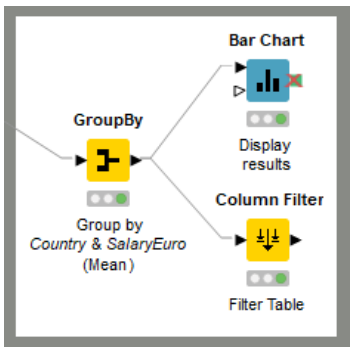


Figura nº51 – Nodos Implementados

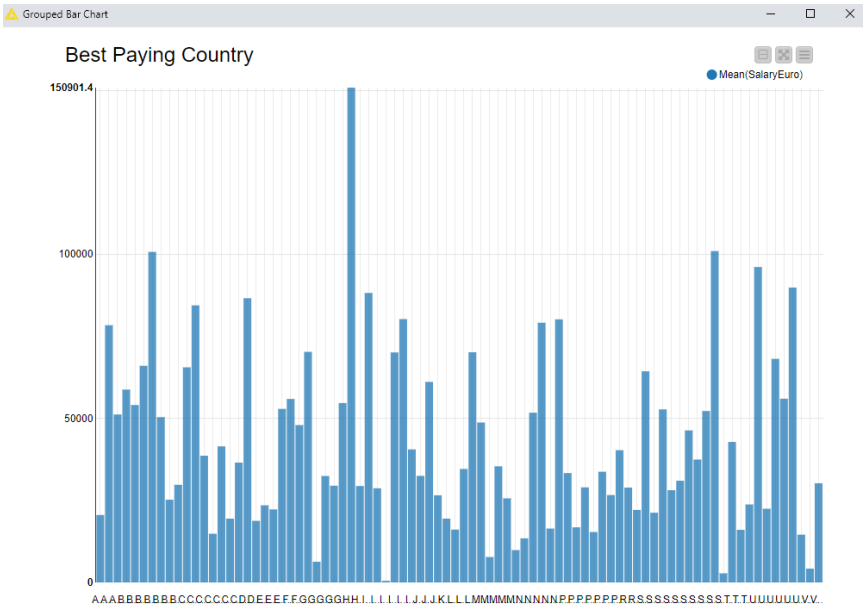
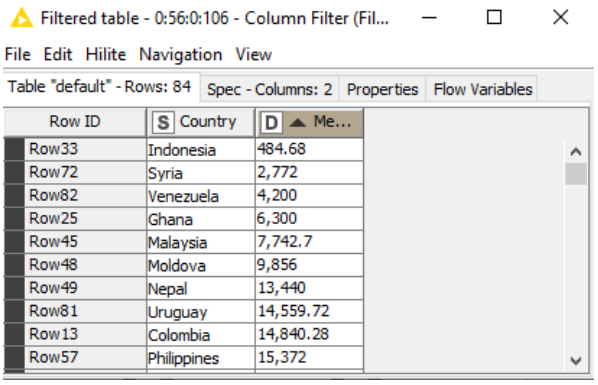


Figura nº52 – ***Bar Chart*** Gerado

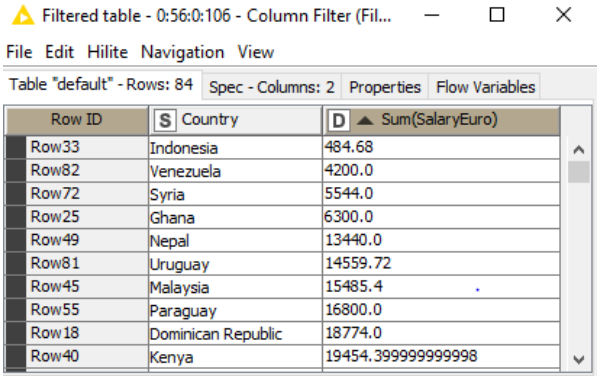


Row ID	Country	Me...
Row33	Indonesia	484.68
Row72	Syria	2,772
Row82	Venezuela	4,200
Row25	Ghana	6,300
Row45	Malaysia	7,742.7
Row48	Moldova	9,856
Row49	Nepal	13,440
Row81	Uruguay	14,559.72
Row13	Colombia	14,840.28
Row57	Philippines	15,372

Figura nº53 – Top de Países com Salário mais Alto

2. Qual o país que paga pior?

À semelhança do *workflow* implementado na tarefa anterior, foi possível observar que o país que paga pior, sendo que no caso é a Indonésia.



Row ID	Country	Sum(SalaryEuro)
Row33	Indonesia	484.68
Row82	Venezuela	4200.0
Row72	Syria	5544.0
Row25	Ghana	6300.0
Row49	Nepal	13440.0
Row81	Uruguay	14559.72
Row45	Malaysia	15485.4
Row55	Paraguay	16800.0
Row18	Dominican Republic	18774.0
Row40	Kenya	19454.399999999998

Figura nº54 – Top de Países com Salário mais Baixo

3. Qual a base de dados primária mais utilizada?

De modo a conseguirmos saber qual a base primária mais utilizada, aplicamos o nodo GroupBy ao nosso workflow, configurando o mesmo de acordo com o que está apresentado nas imagens seguintes.

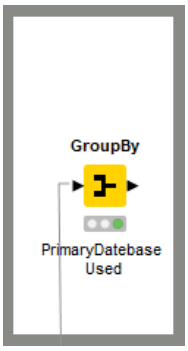


Figura nº55 – Nodo Implementado

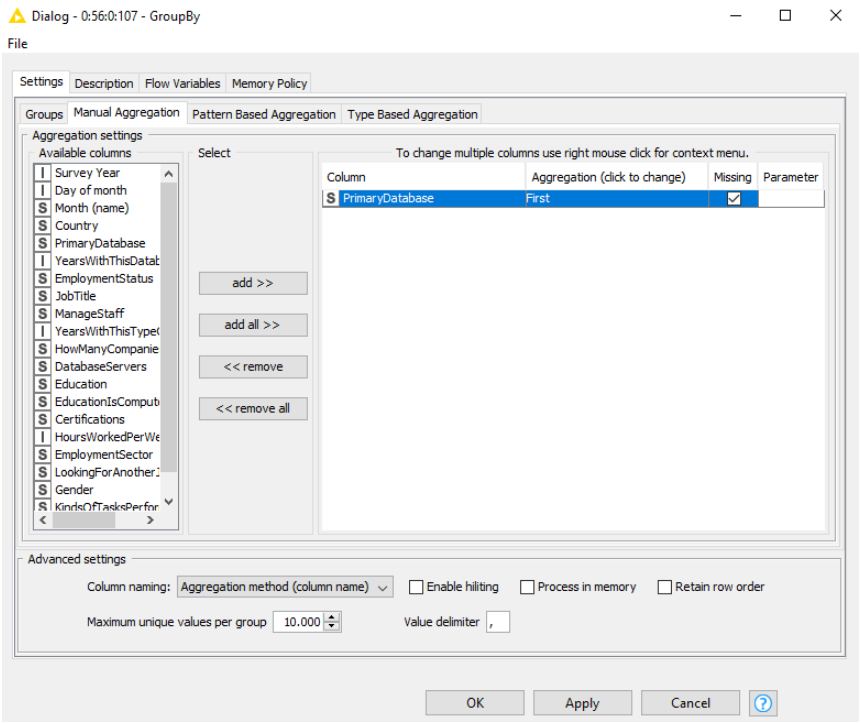


Figura nº55 – Configuração do nodo ***GroupBy***

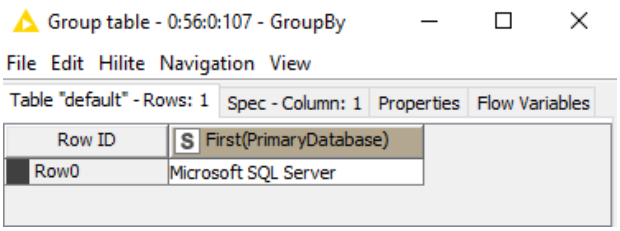


Figura nº56 – Base de Dados Primária mais Utilizada

4. Qual a média de anos de experiência dos colaboradores em cada área?

Foram implementados os nodos ***GropuBy***, ***Bar Chart*** e ***Column Filter*** de modo a conseguirmos responder a esta questão. Pudemos fazer a análise dos resultados obtidos através da leitura da Figura nº59, onde conseguimos ver a média de anos de experiência dos colaboradores em cada área.

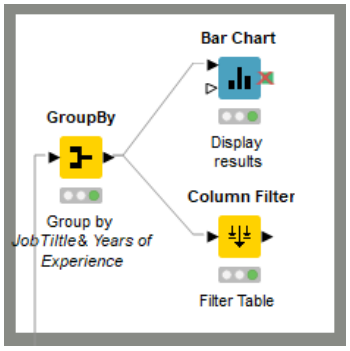


Figura nº57 – Nodos Implementados

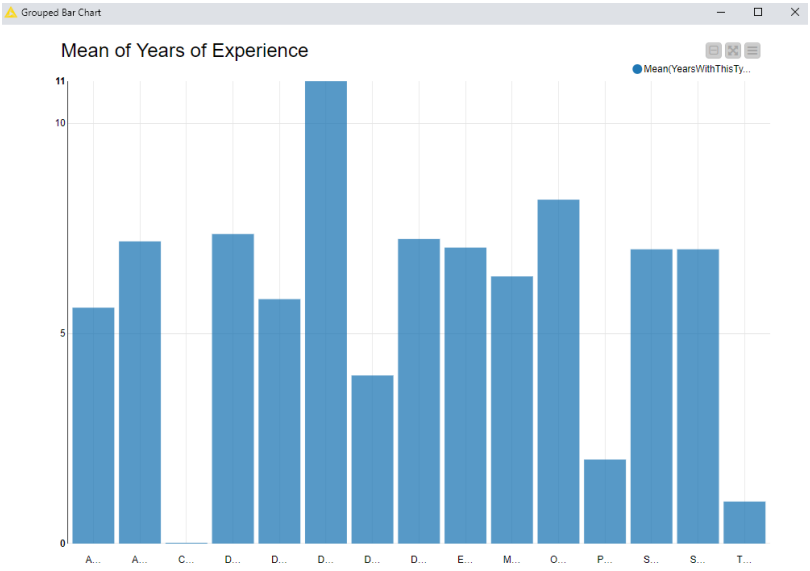
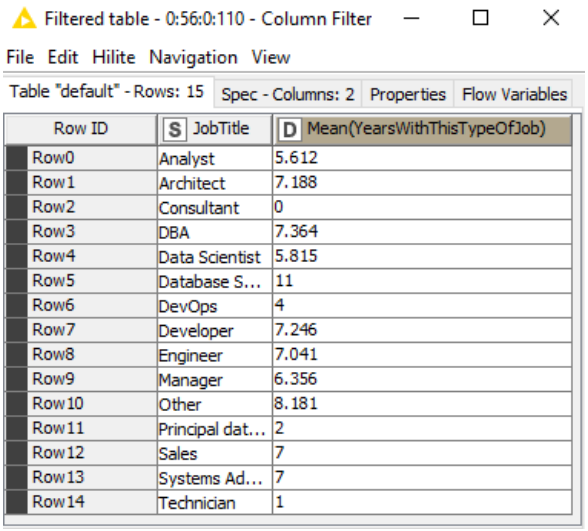


Figura nº58 – ***Bar Chart*** Gerado



Row ID	JobTitle	Mean(YearsWithThisTypeOfJob)
Row0	Analyst	5.612
Row1	Architect	7.188
Row2	Consultant	0
Row3	DBA	7.364
Row4	Data Scientist	5.815
Row5	Database S...	11
Row6	DevOps	4
Row7	Developer	7.246
Row8	Engineer	7.041
Row9	Manager	6.356
Row10	Other	8.181
Row11	Principal dat...	2
Row12	Sales	7
Row13	Systems Ad...	7
Row14	Technician	1

Figura nº59 – Média de Anos de Experiência dos Colabores por Área de Trabalho

5. Qual a percentagem de funcionários em regime de *part-time*? E em *full-time*?

Foram implementados os nodos ***GropuBy***, ***Bar Chart*** de modo a conseguirmos responder a esta questão. Após a análise, conseguimos perceber que a percentagem de funcionário em *part-time* é de 0.3%, em *full-time* é de 95.9% e em *freelancer* é de 3.8%.

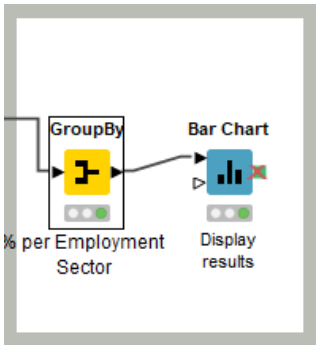


Figura nº60 – Nodos Implementados

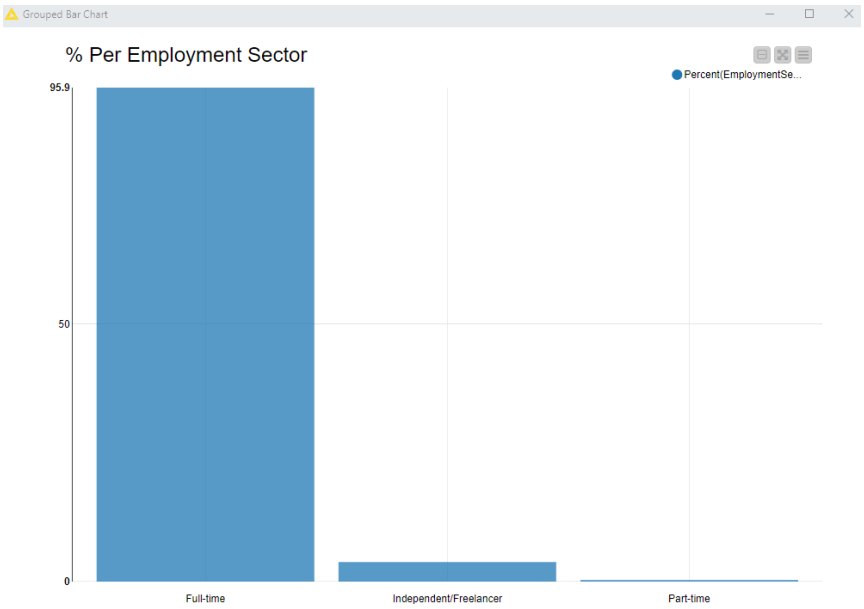


Figura nº61 – ***Bar Chart*** Gerado

6. Um *Team Leader* tem o vencimento mais alto?

Foram implementados os nodos ***GropuBy***, ***Pie/Donut Chart*** de modo a conseguirmos responder a esta questão. Após a análise, conseguimos perceber que um *Team Leader* tem o vencimento mais alto.

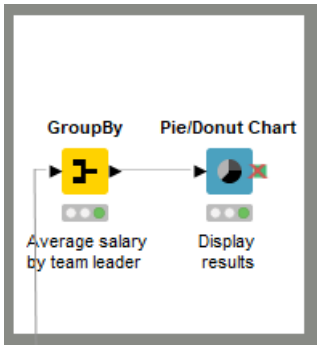


Figura nº62 – Nodos Implementados

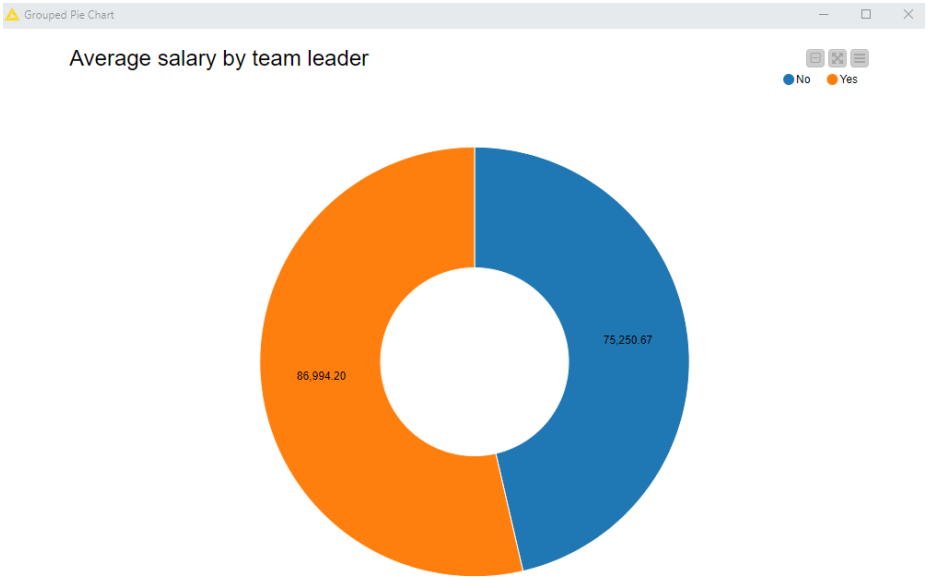


Figura nº63 – ***Pie Chart*** Gerado

7. Qual a média de carga trabalho semanal por país?

Foi implementado o nodo ***Pie/Donut Chart*** de modo a conseguirmos responder a esta questão. A verificação da análise feita pode ver-se na Figura nº64.

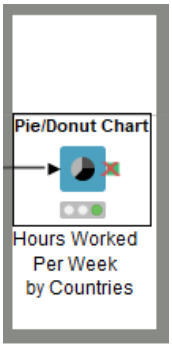


Figura nº64 – Nodo Implementado

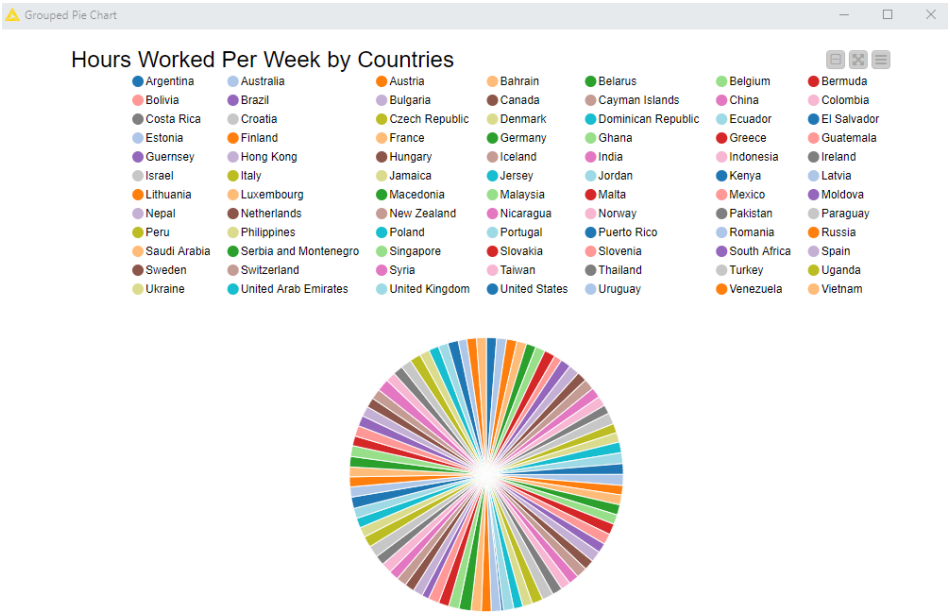


Figura nº65 – ***Pie Chart*** Gerado

8. Maior Salário por Setor e Número de Funcionários Existentes?

Foram implementados os nodos ***GropuBy***, ***Bar Chart*** de modo a conseguirmos responder a esta questão. A verificação da análise feita pode ver-se na Figura nº66.

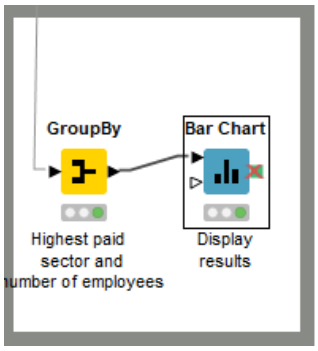


Figura nº66 – Nodos Implementados

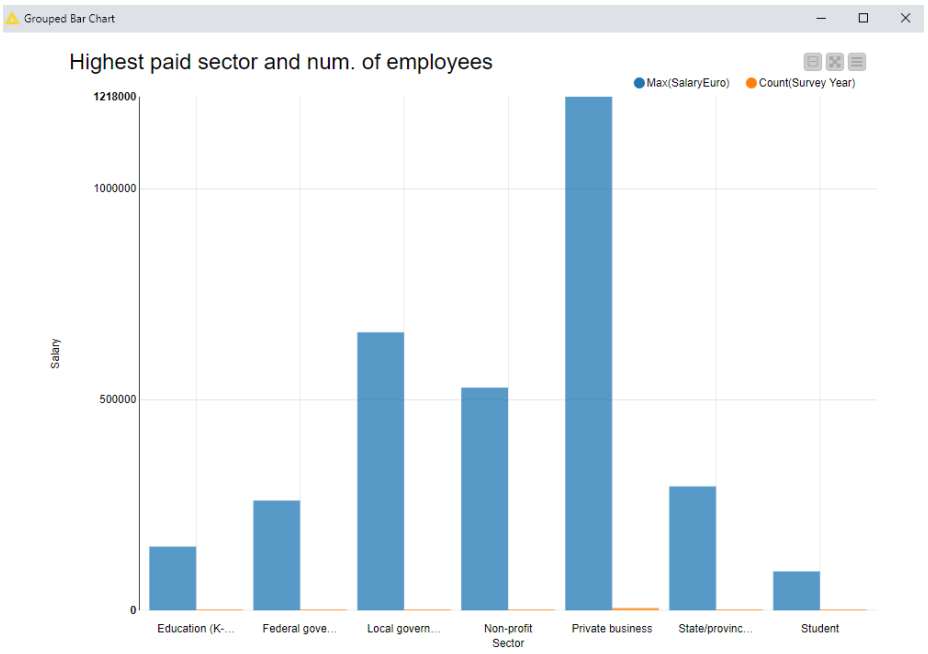


Figura nº67 – ***Bar Chart*** Gerado

9. Existem diferenças salariais por gênero?

Foram implementados os nodos ***Rule-based Row Filter***, ***GropuBy*** de modo a conseguirmos responder a esta questão. Após a análise, conseguimos perceber que não existem grandes diferenças entre gêneros.

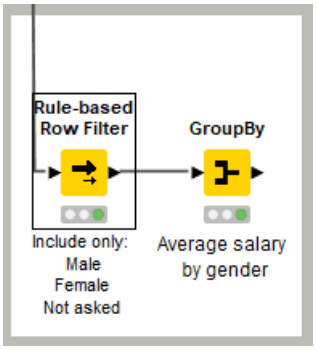


Figura nº68 – Nodos Implementados

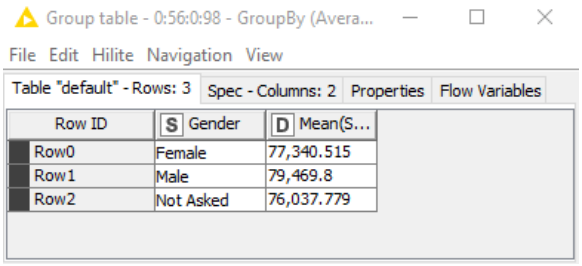


Figura nº69 – *Output* Obtido

10. Análise Geral sobre Portugal

Foram implementados os nodos ***Rule-based Row Filter***, ***GropuBy, Bar Chart*** de modo a conseguirmos responder a esta questão. Após a análise, conseguimos perceber que a maior parte dos trabalhadores têm um curso superior. Existem ainda 6 tipos de profissões desempenhadas por estes.

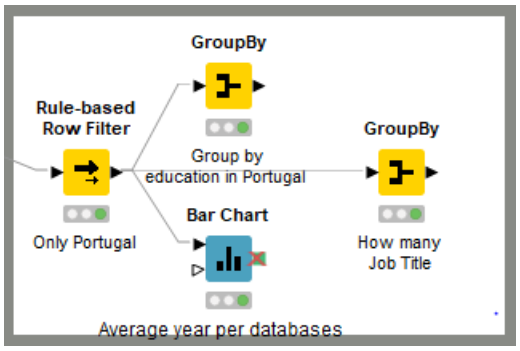


Figura nº70 – Nodos Implementados

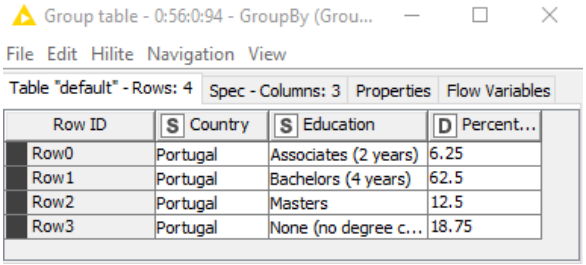


Figura nº71 – *Output* Obtido

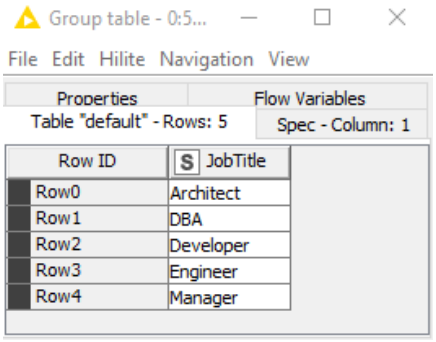


Figura nº72 – *Output* Obtido

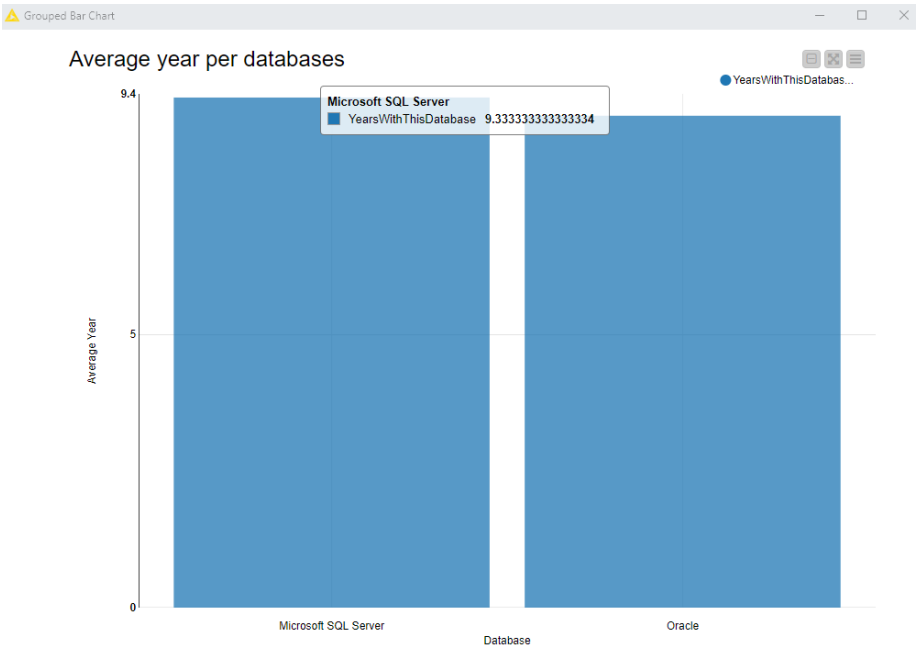


Figura nº73 – ***Bar Chart*** Gerado

11. Média de Salários Anuais

Foi implementado o nodo ***GropuBy*** de modo a conseguirmos responder a esta questão. Após a análise, conseguimos perceber que o salário vai-se mantendo praticamente equivalente ao longo dos 3 anos.

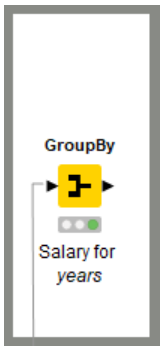
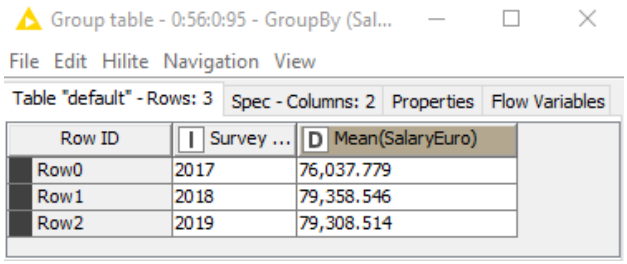


Figura nº74 – Nodo Implementado



Row ID	Survey ...	Mean(SalaryEuro)
Row0	2017	76,037.779
Row1	2018	79,358.546
Row2	2019	79,308.514

Figura nº75 – *Output* Obtido

Tarefa 2. Análise, Tratamento e Exploração de dados do *dataset* (Previsão do Número de Incidentes Rodoviários)

O problema apresentado com este *dataset*, consistia na previsão do número de incidentes rodoviários, uma vez que se trata de um conhecido problema de características estocásticas não-lineares. Tal como é possível observar, este *dataset* foi construído tendo em consideração dados referentes ao número e características dos incidentes rodoviários que ocorreram na cidade de Braga em 2019 (o *dataset* cobre um período que vai desde o dia 01 de Janeiro de 2019 até ao dia 31 de Dezembro do mesmo ano).

a. Carregar, no *Knime*, o *dataset* selecionado

Tal como pudemos observar na imagem que se segue, de modo a conseguir ler o *dataset*, utilizamos e implementados o nodo ***CSV Reader*** no *workflow*.

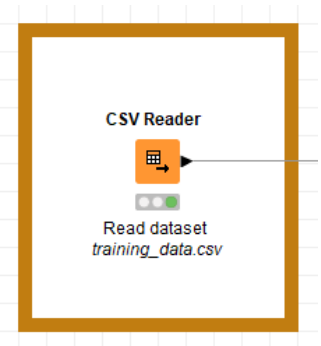


Figura nº76 – Leitura do *dataset*

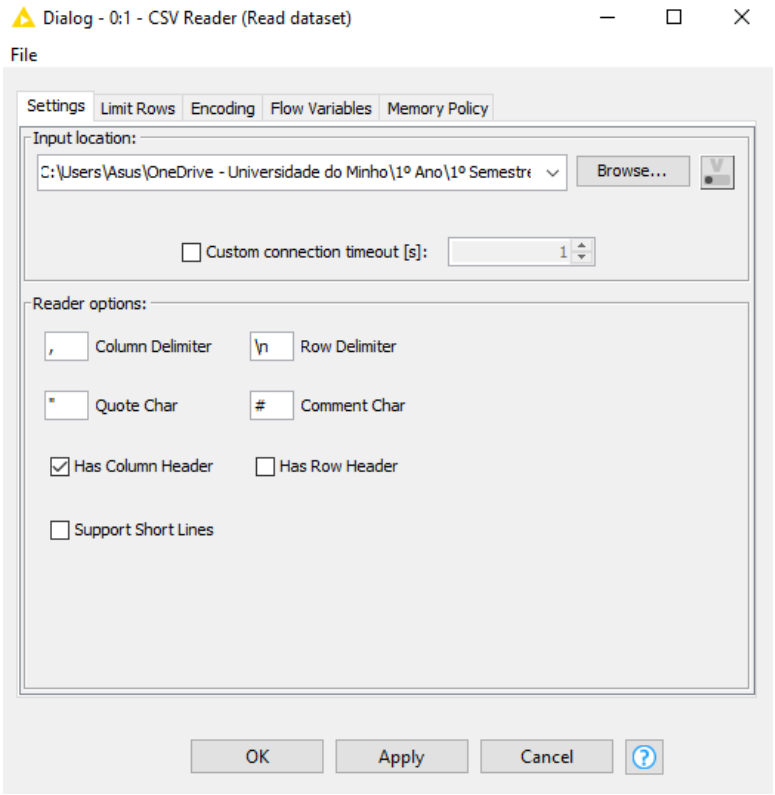


Figura nº77 – Configurações do nodo ***CSV Reader***

File Table - 0:1 - CSV Reader (Read dataset)

File Edit Hilite Navigation View

Table "training_data.csv" - Rows: 5000 | Spec - Columns: 13 | Properties | Flow Variables

Row ID	city_name	magnit...	delay_i...	affecte...	record_date	luminosity	avg_te...	avg_at...	avg_hu...	avg_wi...	avg_pr...	avg_rain	accidents
Row0	Braga	UNDEFINED	0	N103,N103,	2019-02-17 20:00	DARK	13	1,017	87	2	0	chuva fraca	Low
Row1	Braga	MAJOR	401	N101,N103,...	2019-02-14 12:00	LIGHT	17	1,025	45	5	0	Sem Chuva	Medium
Row2	Braga	UNDEFINED	1057	CM1348,CM...	2019-09-03 19:00	DARK	27	1,018	53	4	0	Sem Chuva	Low
Row3	Braga	MAJOR	3612	N101,CM13...	2019-06-18 18:00	LIGHT	16	1,013	100	6	0	Sem Chuva	High
Row4	Braga	MAJOR	498	N14,N101,C...	2019-05-08 13:00	LIGHT	15	1,014	82	6	0	Sem Chuva	Medium
Row5	Braga	MAJOR	1270	CM1348,CM...	2019-05-25 21:00	DARK	19	1,020	82	5	0	Sem Chuva	High
Row6	Braga	UNDEFINED	0	CM1348,CM...	2019-09-16 04:00	DARK	17	1,017	93	2	0	Sem Chuva	None
Row7	Braga	MAJOR	1264	CM1348,CM...	2019-08-15 11:00	LIGHT	26	1,021	73	4	0	Sem Chuva	Medium
Row8	Braga	UNDEFINED	0	N103,N103,	2019-02-21 06:00	DARK	10	1,022	76	5	0	Sem Chuva	Low
Row9	Braga	MAJOR	23804	N14,N103,N...	2019-05-06 17:00	LIGHT	16	1,020	72	2	0	Sem Chuva	Very_High
Row10	Braga	MAJOR	867	CM1348,N1...	2019-09-17 14:00	LIGHT	22	1,016	100	1	0	Sem Chuva	Medium
Row11	Braga	MAJOR	315	CM1348,A1...	2019-10-10 20:00	DARK	16	1,017	82	2	0	Sem Chuva	Low
Row12	Braga	UNDEFINED	0	CM1348,CM...	2019-10-05 13:00	LIGHT	20	1,019	72	3	0	Sem Chuva	None
Row13	Braga	UNDEFINED	0	,	2019-01-19 23:00	DARK	12	1,010	87	6	0	Sem Chuva	None
Row14	Braga	MAJOR	2714	N14,N103,C...	2019-10-17 13:00	LIGHT	17	1,016	82	3	0	Sem Chuva	High
Row15	Braga	MAJOR	1004	CM1348,N1...	2019-09-12 13:00	LIGHT	29	1,025	42	3	0	Sem Chuva	Medium
Row16	Braga	MAJOR	9298	EM569,N10...	2019-11-15 08:00	LIGHT	5	1,012	80	2	0	Sem Chuva	Very_High
Row17	Braga	UNDEFINED	0	CM1348,CM...	2019-07-24 06:00	LIGHT	15	1,015	100	1	0	Sem Chuva	None
Row18	Braga	MAJOR	2059	CM1348,N1...	2019-10-03 15:00	LIGHT	21	1,021	68	2	0	Sem Chuva	High
Row19	Braga	MAJOR	658	N101,N201	2019-04-17 10:00	LIGHT	12	1,007	87	6	0	chuva fraca	None
Row20	Braga	UNDEFINED	0	,	2019-12-23 06:00	DARK	8	1,026	93	3	0	Sem Chuva	None
Row21	Braga	UNDEFINED	0	CM1348,CM...	2019-05-19 13:00	LIGHT	18	1,016	55	6	0	Sem Chuva	Low
Row22	Braga	MAJOR	262	,	2019-05-11 19:00	DARK	23	1,021	68	4	0	Sem Chuva	None
Row23	Braga	UNDEFINED	363	N201,CM13...	2019-07-01 06:00	LIGHT	16	1,020	100	0	0	Sem Chuva	Medium
Row24	Braga	UNDEFINED	0	,	2019-01-21 23:00	DARK	6	1,025	93	1	0	Sem Chuva	None
Row25	Braga	MAJOR	10998	N103,N14,N...	2019-02-06 18:00	DARK	12	1,025	87	1	0	Sem Chuva	Very_High
Row26	Braga	MAJOR	0	CM1348,CM...	2019-06-08 20:00	DARK	14	1,020	82	6	0	Sem Chuva	Medium
Row27	Braga	UNDEFINED	0	CM1348,CM...	2019-09-22 20:00	DARK	15	1,019	93	2	0	Sem Chuva	None
Row28	Braga	MAJOR	1945	CM1348,N1...	2019-11-04 16:00	LIGHT	13	1,010	93	3	0	aguaceiros	High
Row29	Braga	MAJOR	11081	N14,N103,N...	2019-06-05 17:00	LIGHT	15	1,017	67	3	0	Sem Chuva	Very_High
Row30	Braga	UNDEFINED	0	CM1348,CM...	2019-10-22 03:00	DARK	8	1,016	87	3	0	Sem Chuva	None
Row31	Braga	MAJOR	639	N14,CM134...	2019-09-28 18:00	DARK	17	1,019	82	1	0	Sem Chuva	Medium
Row32	Braga	UNDEFINED	123	CM1348,N1...	2019-07-05 10:00	LIGHT	20	1,017	88	4	0	Sem Chuva	Medium
Row33	Braga	UNDEFINED	232	CM1348,CM...	2019-10-19 01:00	DARK	13	1,011	100	6	0	Sem Chuva	Low
Row34	Braga	MODERATE	1018	N14,	2019-12-13 21:00	DARK	12	1,021	82	1	0	Sem Chuva	Medium
Row35	Braga	MAJOR	277	CM1348,CM...	2019-08-10 14:00	LIGHT	22	1,021	73	4	0	Sem Chuva	Low
Row36	Braga	UNDEFINED	0	CM1348,CM...	2019-10-31 23:00	DARK	17	1,024	100	3	0	Sem Chuva	None
Row37	Braga	MAJOR	9454	N103,N14,N...	2019-07-16 17:00	LIGHT	25	1,013	78	2	0	Sem Chuva	Very_High

Figura nº78 – Output Table do *dataset*

b. Aplicar nodos, de modo fazer o Tratamento dos Dados

Com o objetivo de fazer o tratamento dos dados presentes no *dataset*, foram aplicados vários nodos, como por exemplo, nodos para ***extração de datas***, nodos para ***filtrar atributos***, nodos para ***remoção de duplicados***, nodos para ***substituição de Strings***, entre outros. Todos estes nodos implementados para tratamento de dados, encontram-se devidamente evidenciados e explicados nos pontos que se seguem. De salientar que os já referidos nodos, encontram-se dentro de um ***metanode*** criado (***Metanode training data***).

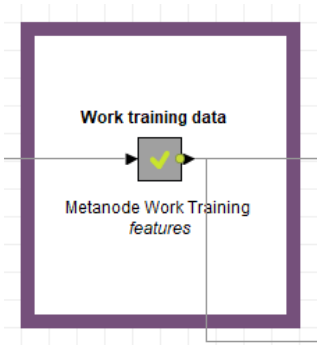


Figura nº79 – ***Metanode*** Implementado para Tratamento de Dados Treino

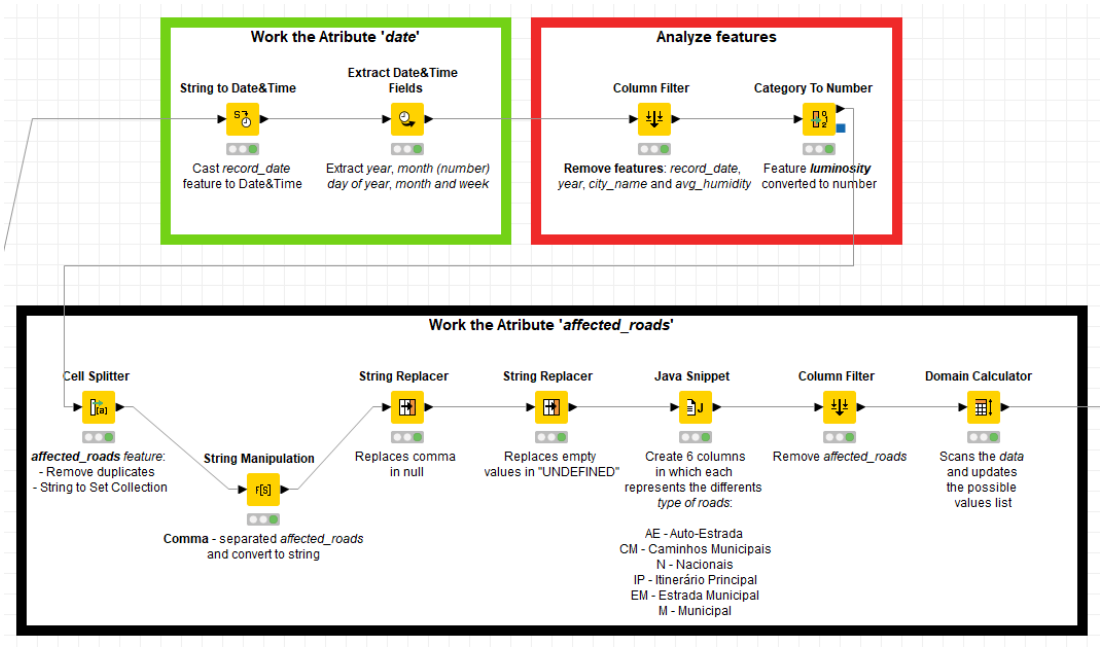


Figura nº80 –*Workfkow* Implementado para Tratamento de Dados Treino

1. Tratar o atributo data

No que ao tratamento de dados diz respeito, o atributo data foi o primeiro a ser tratado. Para isso, foram implementados no *workflow* os nodos ***String to Date&Time*** e ***Extract Date&Time Fields***, uma vez que o nosso objetivo com a implementação destes passava por em primeiro lugar fazer o cast do *record_date* da *feature* para o formato *Date&Time*, de modo a que na aplicação do nodo seguinte, fosse possível extrair o ano, o mês (número), o dia do ano, do mês e da semana. Nas imagens que se seguem conseguimos visualizar as já enunciadas configurações nos respetivos nodos.

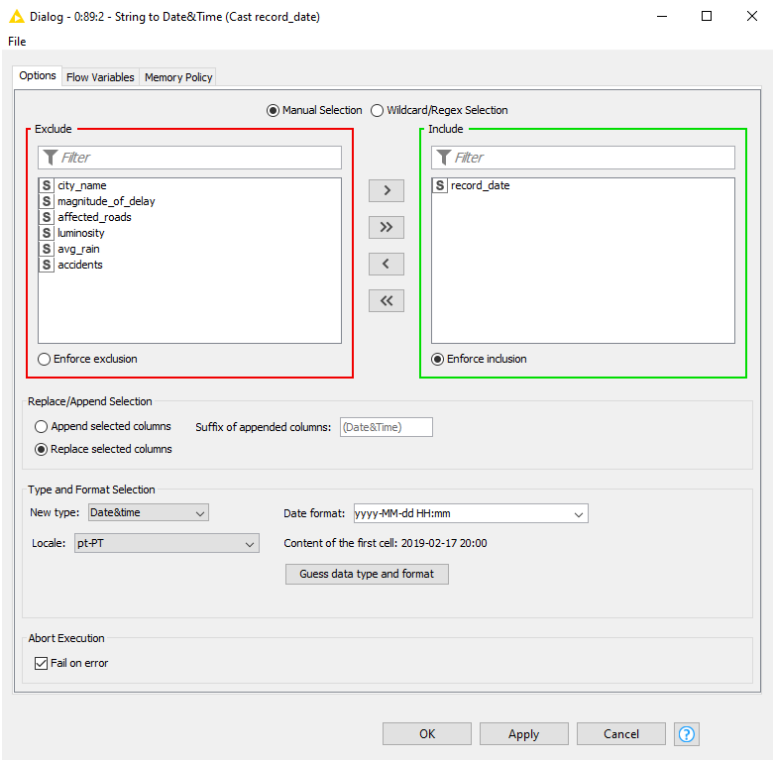


Figura nº81 – Configurações do nodo ***String to Date&Time Fields***

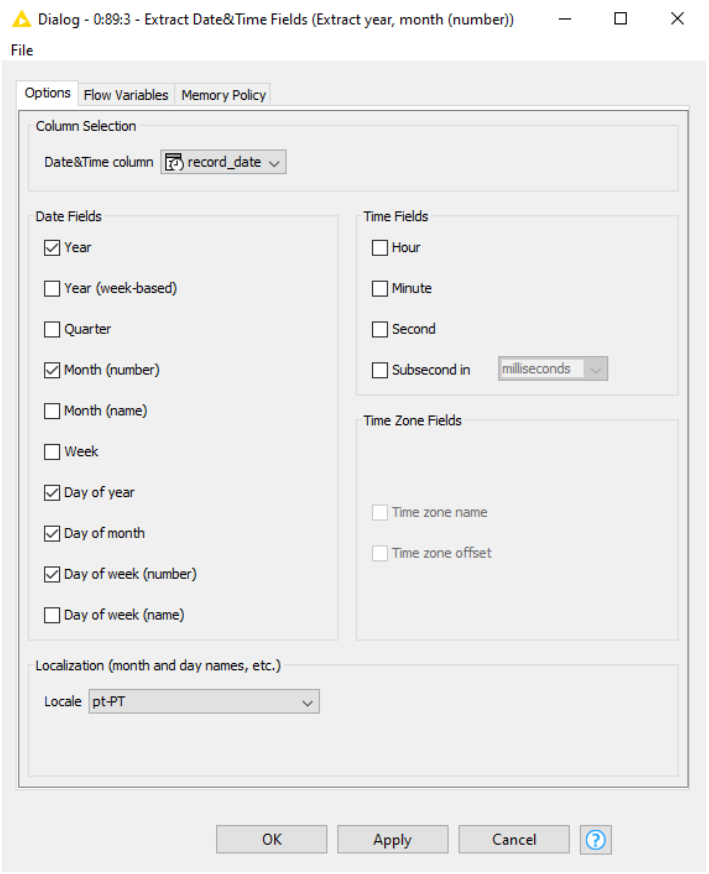


Figura nº82 – Configurações do nodo ***Extract Date&Time Fields***

2. Análise de Atributos

Na "análise" de atributos, aquilo que foi o nosso objetivo na implementação desta abordagem no nosso *workflow*, passou conseguir perceber quais os atributos que trariam maior valor para o nosso conjunto de dados, uma vez que aqueles que na nossa opinião não acrescentavam nada de relevante, foram removidos, recorrendo ao nodo ***Column Filter***. Depois de termos percebido que os atributos *city_name*, *record_date*, *avg_humidity* e *Year*, não acrescentavam valor ao modelo, decidimos tratar os restantes atributos, nomeadamente o *luminosity*. Recorrendo ao nodo ***Category To Number***, decidimos que o atributo *luminosity*, teria um *start value* de 0, um *increment* de 1 e *max. categories* de 100. À semelhança do ponto anterior, as imagens que se seguem evidenciam as já enunciadas configurações.

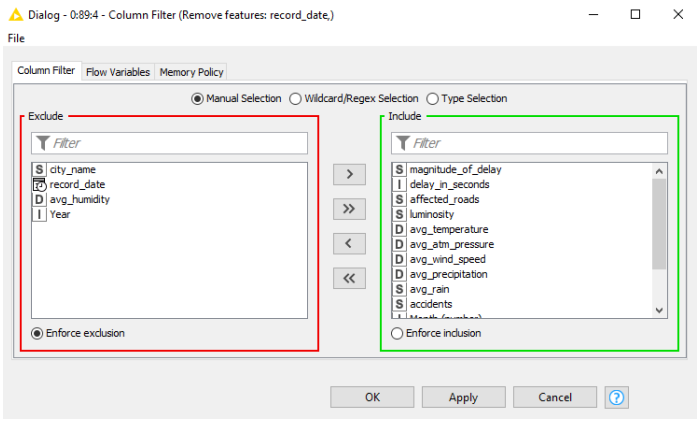


Figura nº83 – Configurações do nodo ***Column Filter***

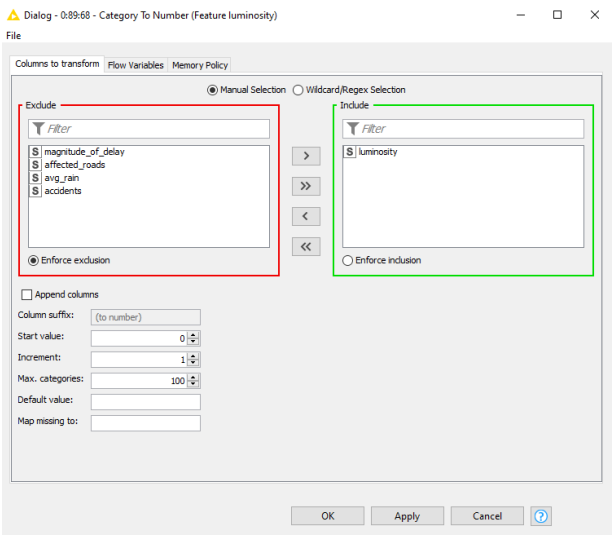


Figura nº84 – Configurações do nodo ***Category To Number***

3. Tratar o atributo estradas afetadas

Para tratarmos do atributo *affected_roads*, tivemos de aplicar na totalidade 6 nodos, uma vez que se tratava do atributo mais complexo para o modelo, pelo facto de como eram apresentados os dados no mesmo. Deste modo, a primeira coisa que fizemos foi a remoção de duplicados e fizemos ainda um *String to Set Collection*, recorrendo nodo ***Cell Splitter***, evidenciado no *workflow*.

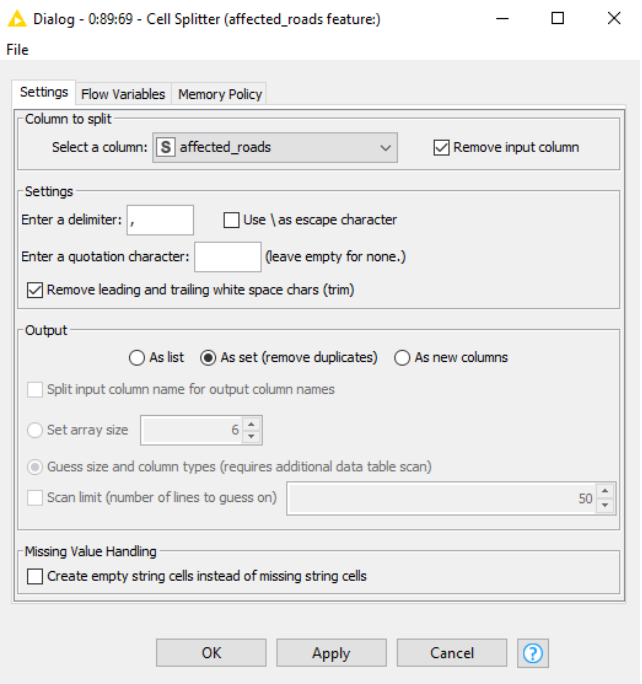


Figura nº85 – Configurações do nodo ***Cell Spliter***

De acordo com o que é apresentado na imagem que se segue, pudemos verificar que com a implementação do nodo ***String Manipulation***, através das vírgulas, separamos as estradas afetadas, convertendo as mesmas para String.

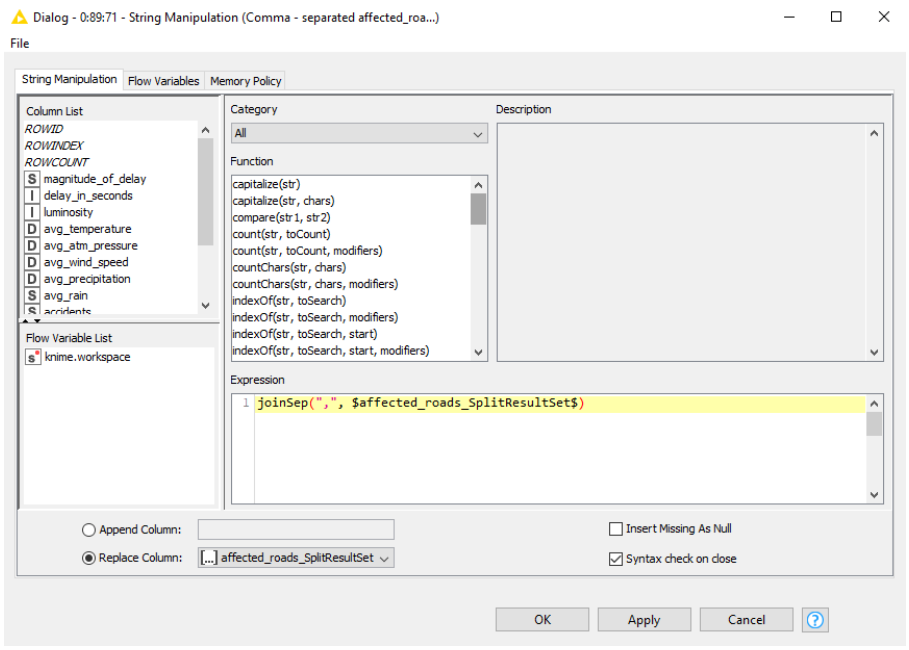


Figura nº86 – Configurações do nodo ***String Manipulation***

Recorrendo ao primeiro nodo ***String Replacer***, fizemos a substituição das vírgulas para *null*, enquanto que no segundo nodo ***String Replacer***, fizemos a substituição dos campos vazios/nulos para *UNDEFINED*, de modo a ser mais fácil identificar os mesmos.

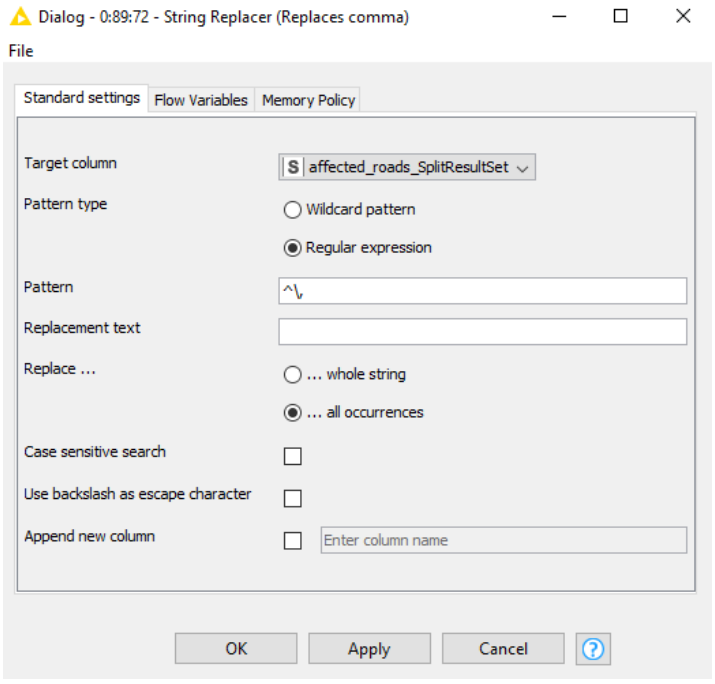


Figura nº87 – Configurações do 1º nodo ***String Replacer***

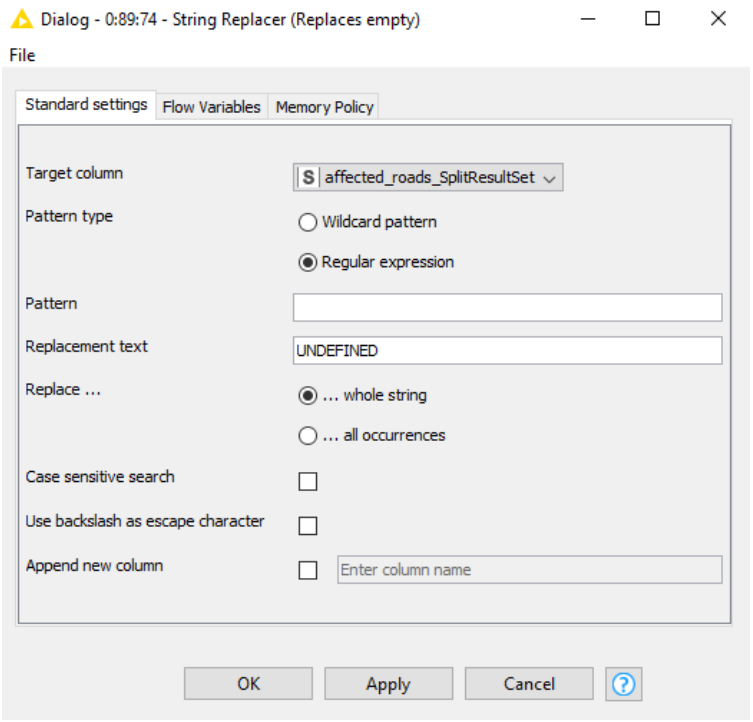


Figura nº88 – Configurações do 2º nodo ***String Replacer***

Com a implementação do nodo ***Java Snippet***, e uma vez que o atributo *affected_roads*, contém informação referente a várias estradas, desde estradas nacionais, autoestradas, itinerários principais, entre outras, conseguimos criar 6 colunas, sendo que cada uma delas representa 1 tipo de estrada. Deste modo, no nosso modelo, ficámos com os seguintes atributos:

- AE – Autoestrada
- CM – Caminhos Municipais
- N – Nacionais
- IP – Itinerário Principal
- EM – Estrada Municipal
- M – Municipal

Pudemos observar as enunciadas configurações na imagem que se segue.

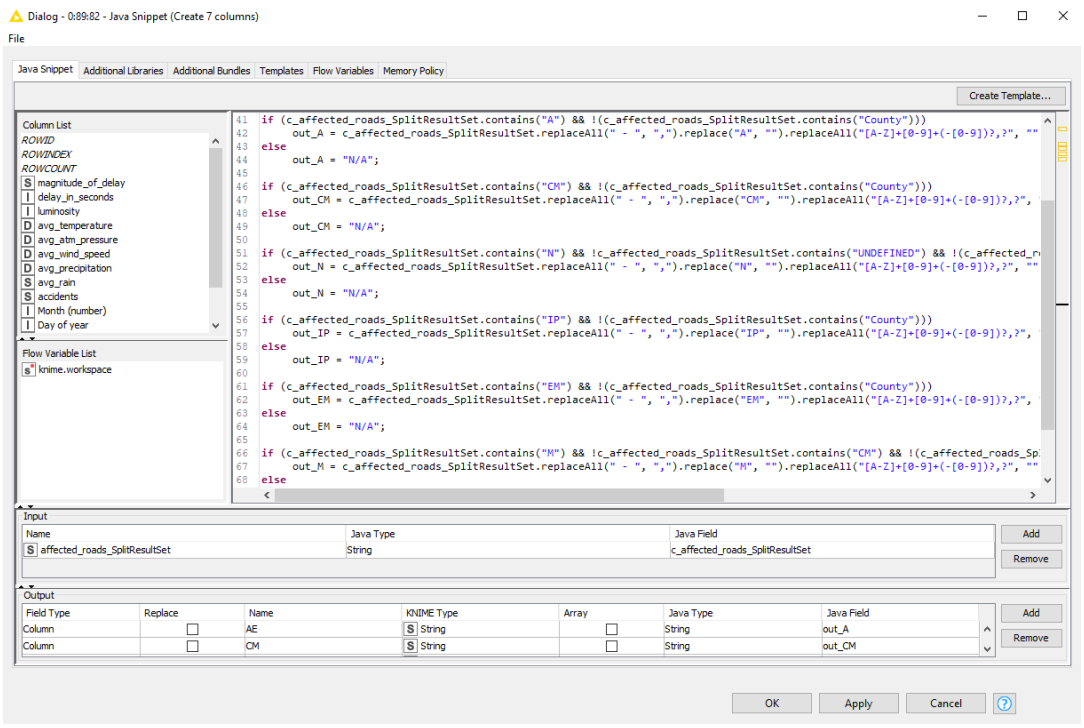


Figura nº89 – Configurações do nodo ***Java Snippet***

Uma vez que criámos 6 atributos novos ao modelo, cada um deles referente a um único tipo de estrada, recorrendo ao ***Column Filter***, removemos o atributo *affected_roads*, uma vez que para o modelo, é extremamente difícil analisar e trabalhar os dados sobre este, e por isso mesmo é que foram criados os 6 atributos no nodo anterior, pois o objetivo passava pela remoção do mesmo.

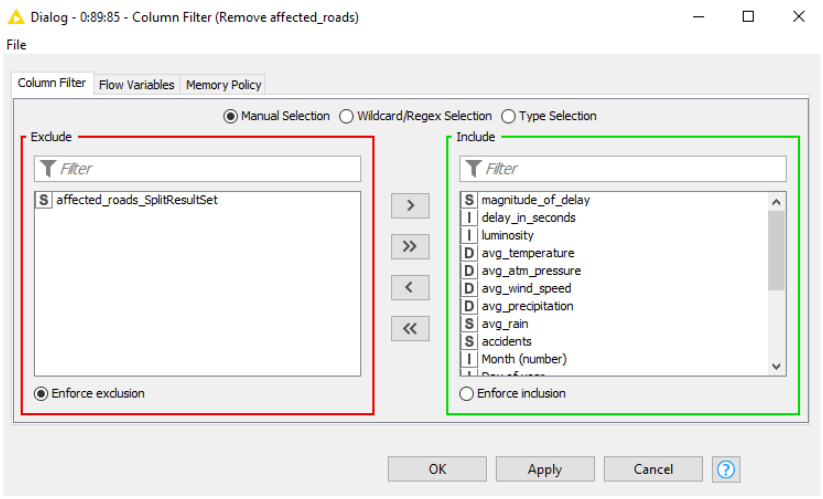


Figura nº90 – Configurações do nodo ***Column Filter***

Por fim, com a implementação do nodo ***Domain Calculator***, foi-nos possível fazer uma análise de dados e atualizações dos mesmos, para a lista de dados selecionada.

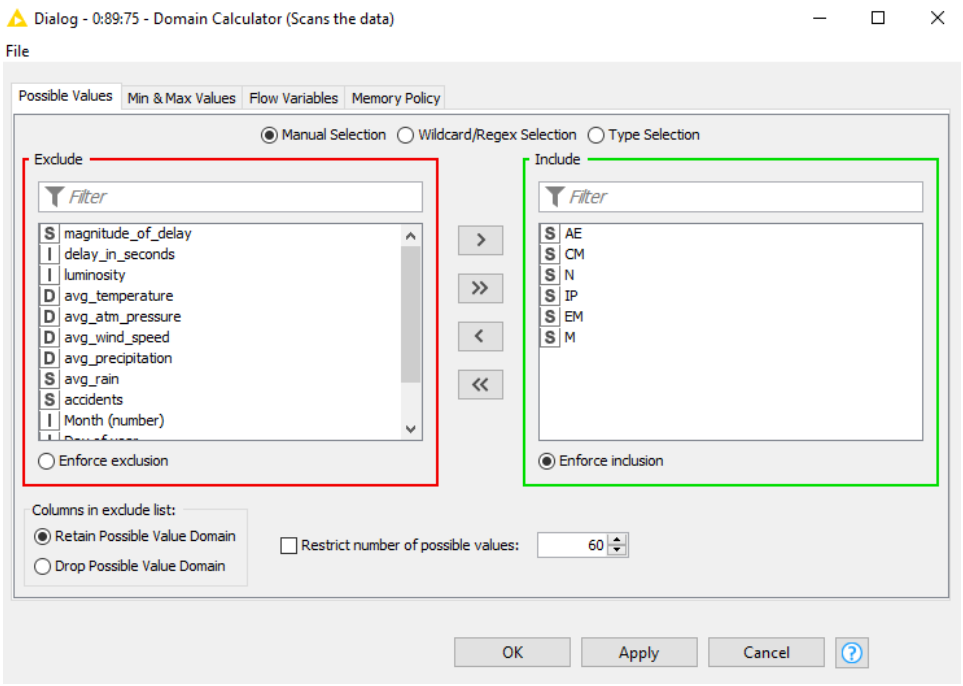


Figura nº91 – Configurações do nodo ***Domain Calculator***

c. Aplicar nodos, de modo fazer a Análise de *Features* do Modelo

Com o objetivo de fazer a análise de dados, foram aplicados vários nodos, de modo a conseguir perceber quais seriam as *features* ideias para o nosso modelo. Todos estes nodos implementados no *workflow*, encontram-se dentro de um ***metanode*** criado (***Metanode Analyze Features***), e estão devidamente explicados neste ponto relatório prático.

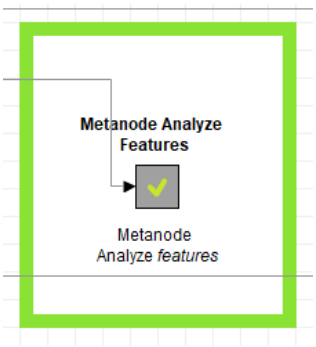


Figura nº92 – ***Metanode*** Implementado para Análise de Dados

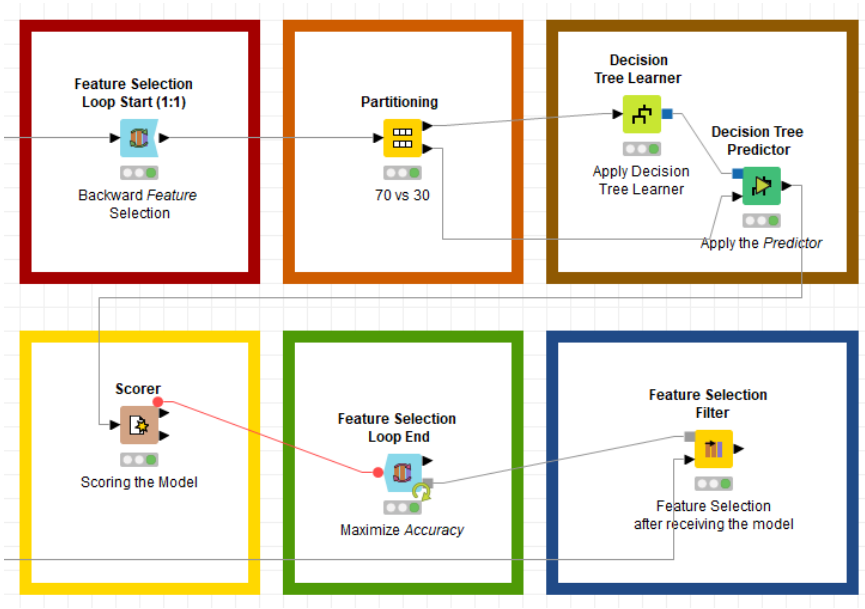


Figura nº93 –*Workfkow* Implementado para Análise de Dados

De modo a conseguirmos fazer a *backward feature selection*, implementamos no *workflow* o nodo ***Feature Selection Loop Start***. Na configuração deste nodo apenas excluímos o atributo '*accidents'*.

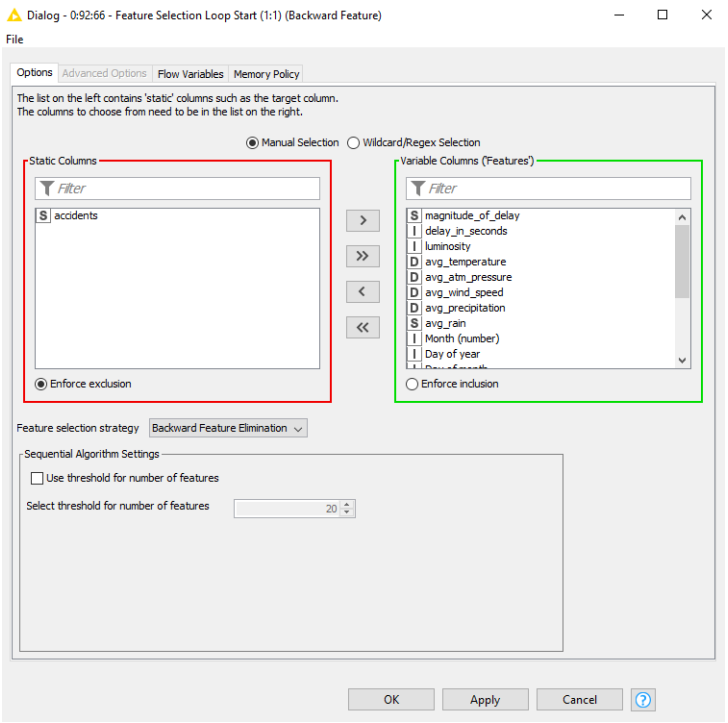


Figura nº94 – Configurações do nodo ***Feature Selection Loop Start***

Recorrendo ao nodo ***Partitioning***, foi feito o particionamento do conjunto de dados, de acordo com as configurações apresentadas na imagem seguinte.

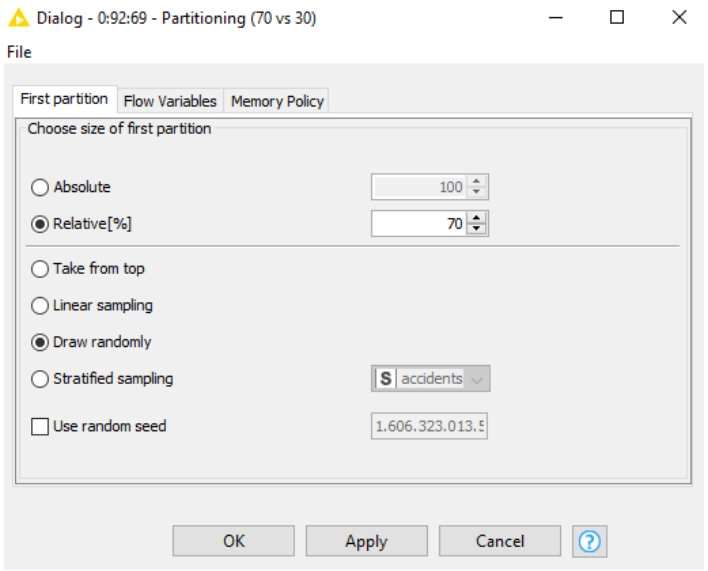


Figura nº95 – Configurações do nodo ***Partitioning***

No que à configuração do nodo ***Decision Tree Learner*** implementado, é importante salientar que a *Quality Measure* selecionada foi a ***Gini index***, ***sem pruning*** no *Pruning method*. As restantes configurações efetuadas no nodo podem ser observadas na imagem seguinte.

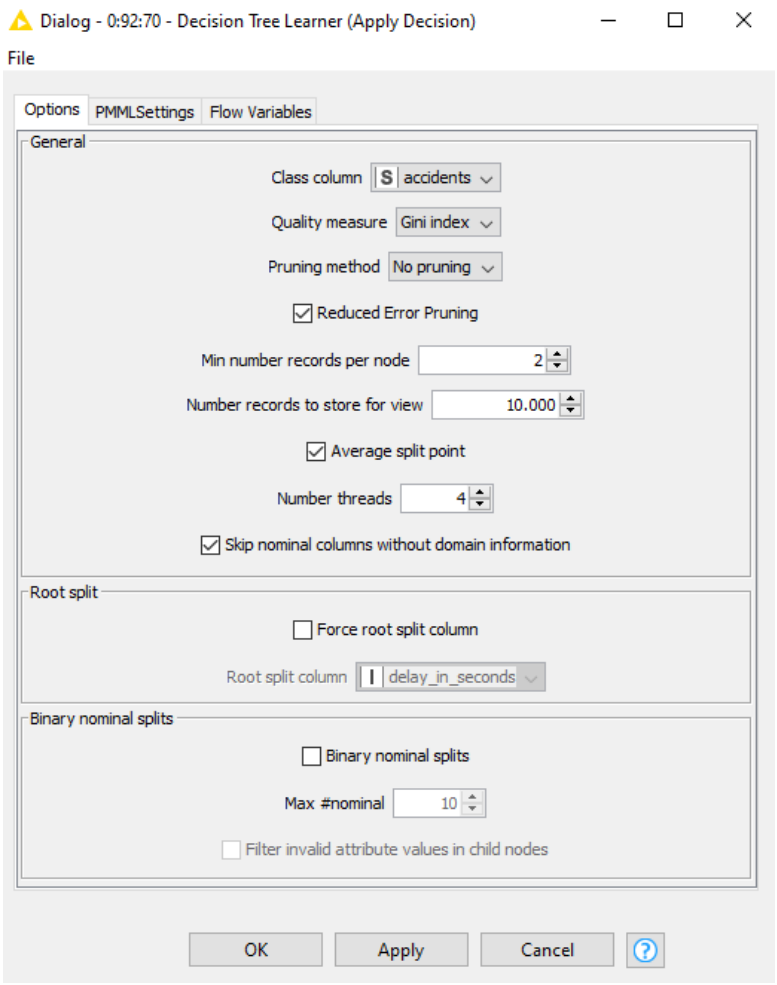


Figura nº96 – Configurações do nodo ***Decision Tree Learner***

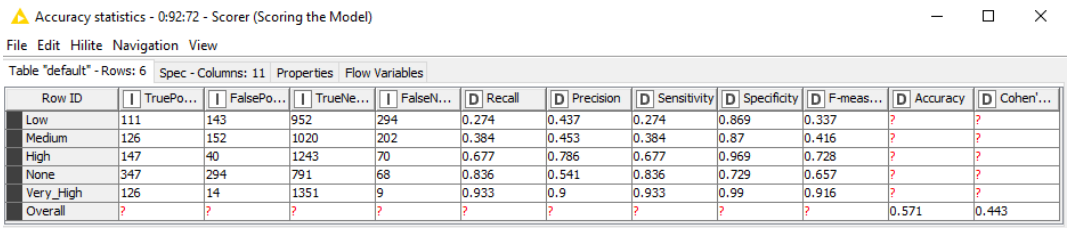
Confusion matrix - 0:92:72 - Scorer (Scoring the Model)

File Edit Hilite Navigation View

Table "spec_name" - Rows: 5 | Spec - Columns: 5 | Properties | Flow Variables

Row ID	Low	Medium	High	None	Very_High
Low	111	77	2	215	0
Medium	98	126	28	76	0
High	8	45	147	3	14
None	37	30	1	347	0
Very_High	0	0	9	0	126

Figura nº97 – *Confusion Matrix* Obtida no nodo ***Scorer***

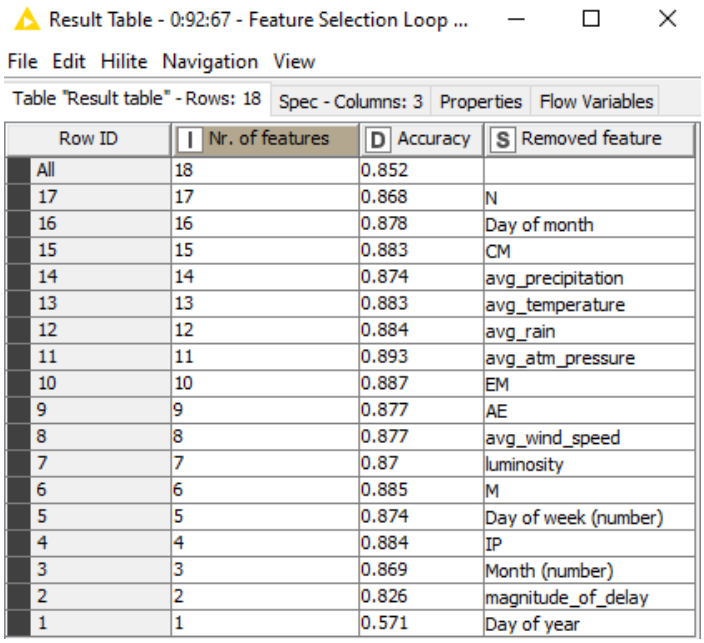

Accuracy statistics - 0:92:72 - Scorer (Scoring the Model)

File Edit Hilite Navigation View

Table "default" - Rows: 6 | Spec - Columns: 11 | Properties | Flow Variables

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
Low	111	143	952	294	0.274	0.437	0.274	0.869	0.337	?	?
Medium	126	152	1020	202	0.384	0.453	0.384	0.87	0.416	?	?
High	147	40	1243	70	0.677	0.786	0.677	0.969	0.728	?	?
None	347	294	791	68	0.836	0.541	0.836	0.729	0.657	?	?
Very_High	126	14	1351	9	0.933	0.9	0.933	0.99	0.916	?	?
Overall	?	?	?	?	?	?	?	?	?	0.571	0.443

Figura nº98 – *Accuracy Statistics* Obtidas no nodo ***Scorer***


Result Table - 0:92:67 - Feature Selection Loop ...

File Edit Hilite Navigation View

Table "Result table" - Rows: 18 | Spec - Columns: 3 | Properties | Flow Variables

Row ID	Nr. of features	Accuracy	Removed feature
All	18	0.852	
17	17	0.868	N
16	16	0.878	Day of month
15	15	0.883	CM
14	14	0.874	avg_precipitation
13	13	0.883	avg_temperature
12	12	0.884	avg_rain
11	11	0.893	avg_atm_pressure
10	10	0.887	EM
9	9	0.877	AE
8	8	0.877	avg_wind_speed
7	7	0.87	luminosity
6	6	0.885	M
5	5	0.874	Day of week (number)
4	4	0.884	IP
3	3	0.869	Month (number)
2	2	0.826	magnitude_of_delay
1	1	0.571	Day of year

Figura nº99 – *Result Table* Obtida no nodo ***Feature Selection Loop End***

Por fim, recorrendo ao nodo ***Feature Selection Filter*** implementado no *workflow*, foi feita a seleção de features depois de receber o modelo.

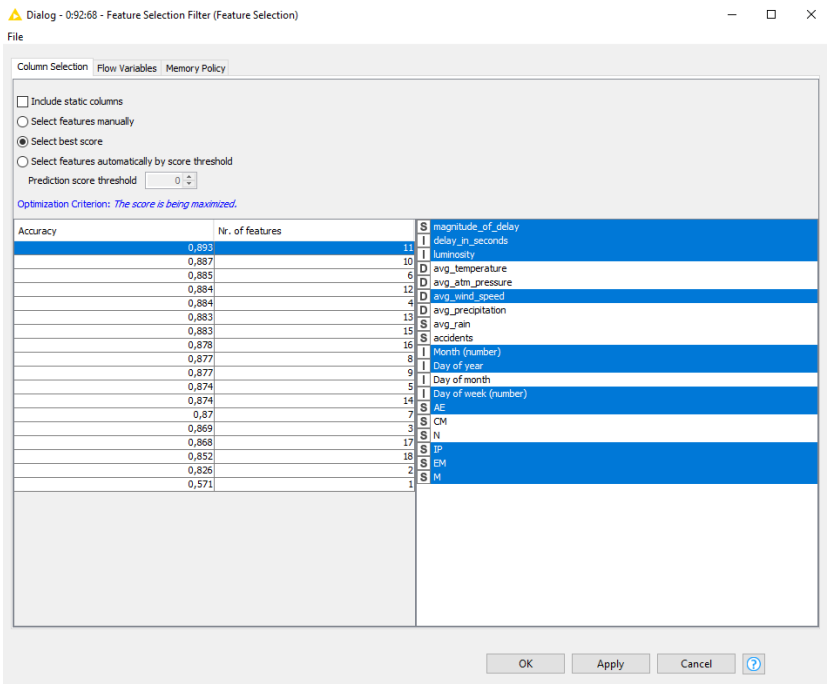
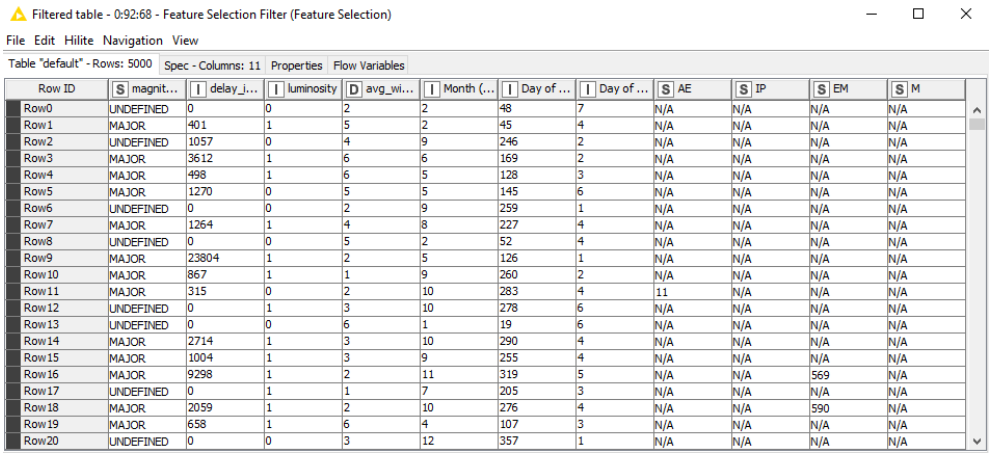


Figura nº100 – Configurações do nodo ***Feature Selection Filter***



Row ID	magnit...	delay_i...	luminosity	avg_wi...	Month (...	Day of ...	Day of ...	AE	IP	EM	M
Row0	UNDEFINED	0	0	2	2	48	7	N/A	N/A	N/A	N/A
Row1	MAJOR	401	1	5	2	45	4	N/A	N/A	N/A	N/A
Row2	UNDEFINED	1057	0	4	9	246	2	N/A	N/A	N/A	N/A
Row3	MAJOR	3612	1	6	6	169	2	N/A	N/A	N/A	N/A
Row4	MAJOR	498	1	6	5	128	3	N/A	N/A	N/A	N/A
Row5	MAJOR	1270	0	5	5	145	6	N/A	N/A	N/A	N/A
Row6	UNDEFINED	0	0	2	9	259	1	N/A	N/A	N/A	N/A
Row7	MAJOR	1264	1	4	8	227	4	N/A	N/A	N/A	N/A
Row8	UNDEFINED	0	0	5	2	52	4	N/A	N/A	N/A	N/A
Row9	MAJOR	23804	1	2	5	126	1	N/A	N/A	N/A	N/A
Row10	MAJOR	867	1	1	9	260	2	N/A	N/A	N/A	N/A
Row11	MAJOR	315	0	2	10	283	4	11	N/A	N/A	N/A
Row12	UNDEFINED	0	1	3	10	278	6	N/A	N/A	N/A	N/A
Row13	UNDEFINED	0	0	6	1	19	6	N/A	N/A	N/A	N/A
Row14	MAJOR	2714	1	3	10	290	4	N/A	N/A	N/A	N/A
Row15	MAJOR	1004	1	3	9	255	4	N/A	N/A	N/A	N/A
Row16	MAJOR	9298	1	2	11	319	5	N/A	N/A	569	N/A
Row17	UNDEFINED	0	1	1	7	205	3	N/A	N/A	N/A	N/A
Row18	MAJOR	2059	1	2	10	276	4	N/A	N/A	590	N/A
Row19	MAJOR	658	1	6	4	107	3	N/A	N/A	N/A	N/A
Row20	UNDEFINED	0	0	3	12	357	1	N/A	N/A	N/A	N/A

Figura nº101 – *Result Table* Obtida no nodo ***Feature Selection Filter***

d. Aplicar nodos, de modo fazer o *Tuning* do Modelo

Com o objetivo de fazer o *tuning* aos dados, foram aplicados vários nodos, de modo a conseguir perceber quais seriam as configurações ideias para o nosso modelo. Todos estes nodos implementados no *workflow*, encontram-se dentro de um ***metanode*** criado (***Metanode Tuning***), e estão devidamente explicados neste ponto relatório prático.

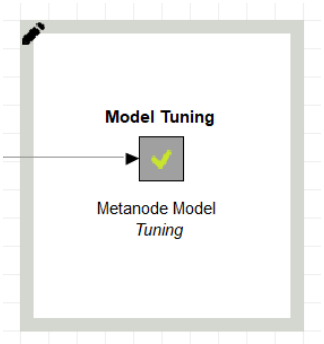


Figura nº102 – ***Metanode*** Implementado para Tuning de Dados

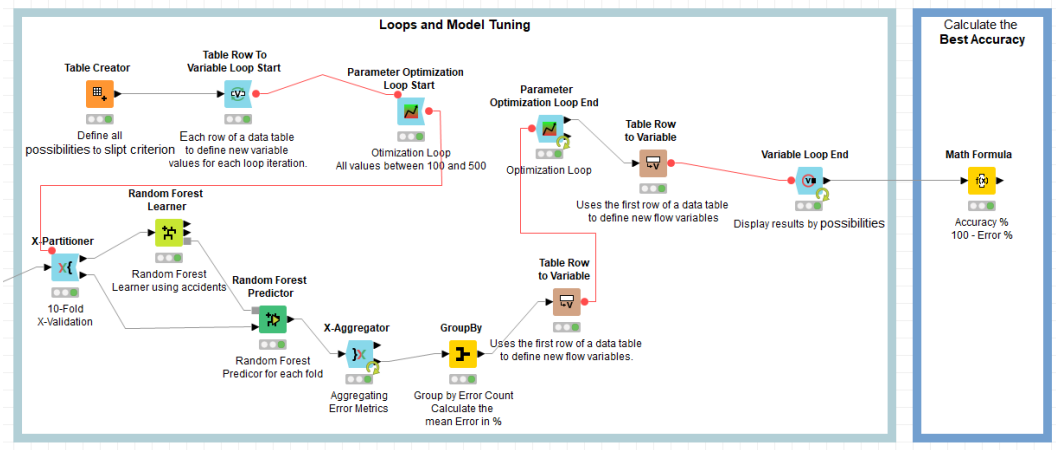


Figura nº103 –*Workfkow* Implementado para Tuning de Dados

Na imagem seguinte é possível observar as configurações feitas no nodo ***Table Creator***, sendo que o objetivo do mesmo é definir todas as possibilidades para o *Split Criterion*.

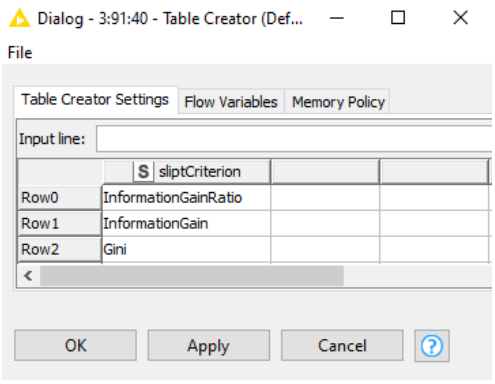


Figura nº104 – Configurações do nodo ***Table Creator***

O nodo ***Table Row To Variable Loop Start*** foi implementado no *workflow* com o objetivo de que em cada linha de uma tabela, fosse definida uma nova variável com valores para cada iteração do *loop*.

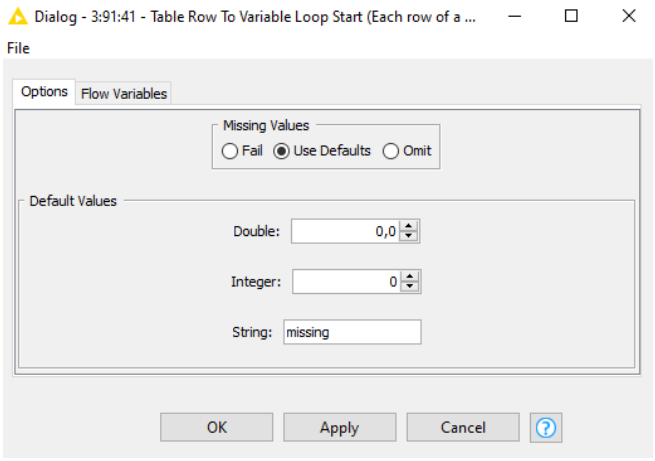


Figura nº105 – Configurações do nodo ***Table Row To Variable Loop Start***

De acordo com as configurações referentes ao nodo ***Parameter Optimization Loop Start***, conseguimos perceber que este nodo foi implementado no *workflow* com o objetivo de fazer a otimização do loop para todos os valores entre 100 e 200.

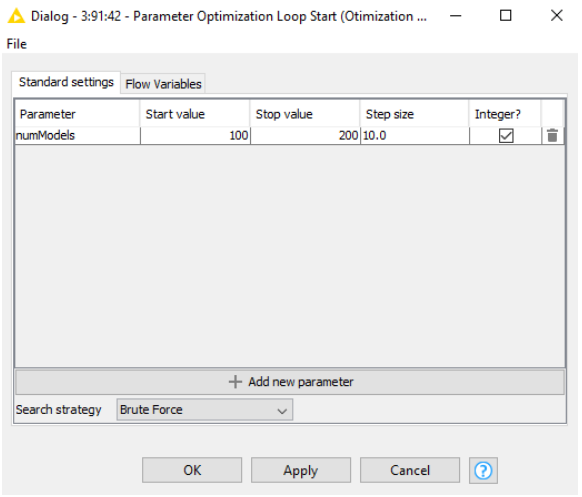


Figura nº106 – Configurações do nodo ***Parameter Optimization Loop Start***

Para fazer uma *X-Validation*, implementamos o nodo ***X-Partitioner*** no nosso *workflow*. Para fazer a agregação, implementamos o ***X-Aggregator***.

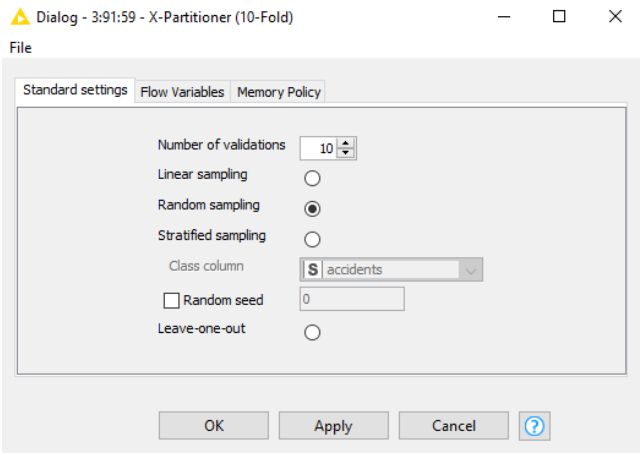


Figura nº107 – Configurações do nodo ***X-Partitioner***

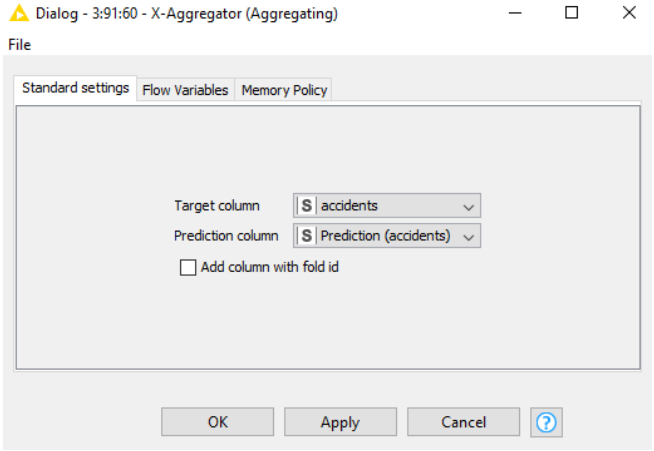


Figura nº108.– Configurações do nodo ***X-Aggregator***

O nodo ***Table Row to Variable*** foi implementado com o objetivo de utilizar os primeiros dados da tabela, de modo a definir novas *flow variables*.

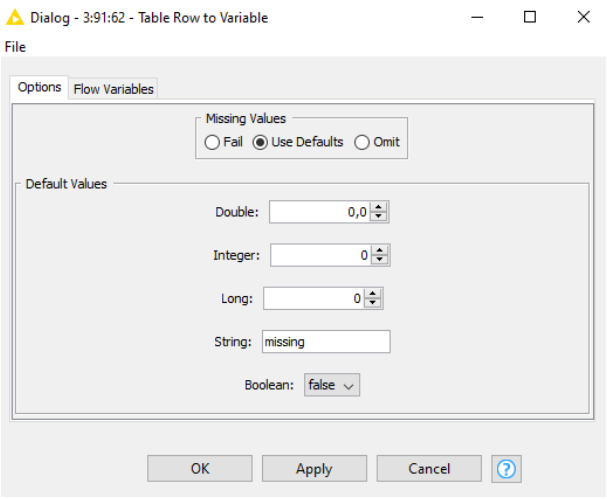


Figura nº109 – Configurações do nodo ***Table Row to Variable***

De modo a otimizar os *loops*, foi implementado o nodo ***Parameter Optimization Loop End*** no *workflow*, sendo que a *flow variable* na função objetivo era a média do erro, sendo que era pretendido a minimização da mesma.

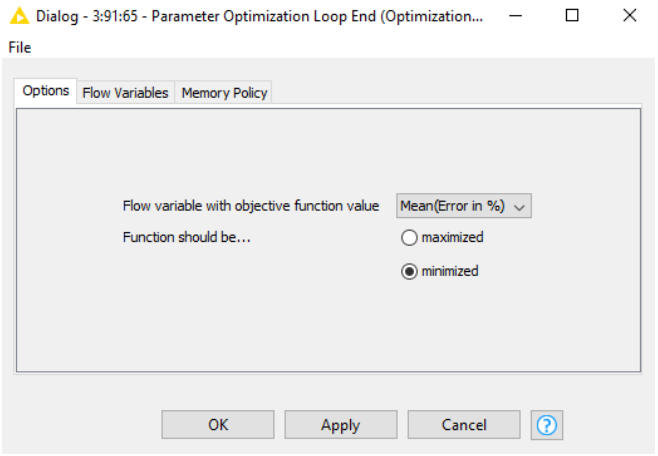


Figura nº110 – Configurações do nodo ***Parameter Optimization Loop End***

À semelhança do outro nodo ***Table Row to Variable***, este foi igualmente implementado com o objetivo de utilizar os primeiros dados da tabela, de modo a definir novas *flow variables*.

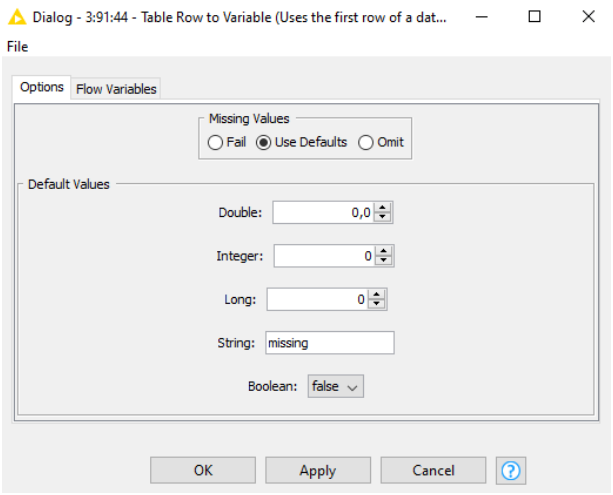


Figura nº111 – Configurações do nodo ***Table Row to Variable***

Este nodo (***Variable Loop End***) foi implementado com o intuito de fazer o display dos resultados de todas as opções possíveis para o modelo a implementar.

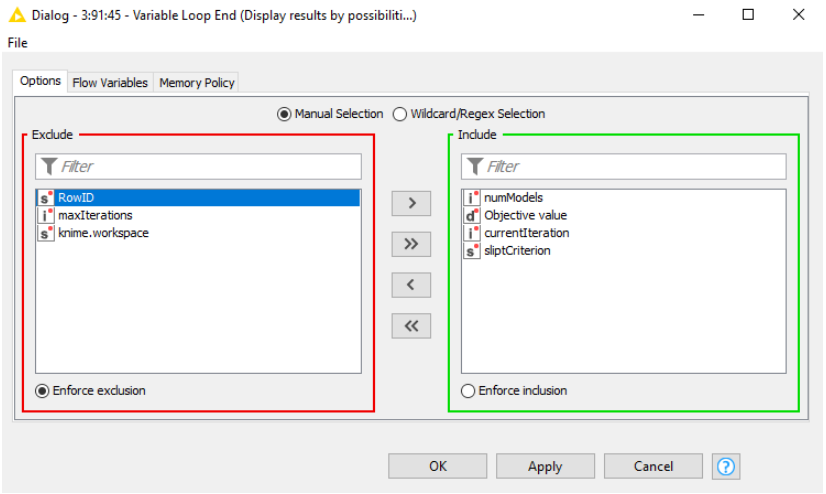


Figura nº112 – Configurações do nodo ***Variable Loop End***

De acordo com as configurações apresentadas na imagem seguinte, é possível observar que o nodo ***Math Formula*** foi implementado com o objetivo de conseguirmos perceber qual é a nossa accuracy numa escala percentual.

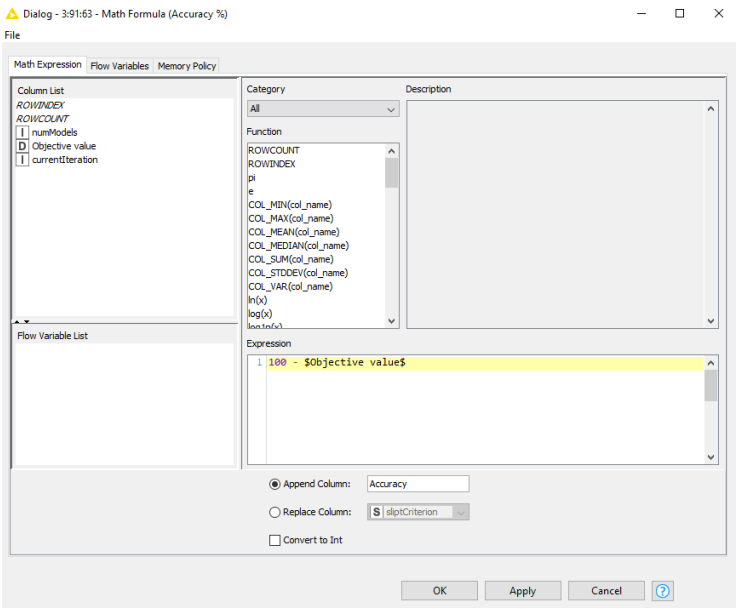


Figura nº113 – Configurações do nodo ***Math Formula***

e. Treino e Resultados Finais do Modelo

À semelhança do que foi feito para o tratamento dos dados presentes no *dataset* (treinar o conjunto de dados), foram aplicados vários nodos, como por exemplo, nodos para ***extração de datas***, nodos para ***filtrar atributos***, nodos para ***remoção de duplicados***, nodos para ***substituição de Strings***, entre outros, mas agora direcionado para todos dados presentes no conjunto de teste. De salientar que os já referidos nodos, semelhantes aos aplicados no conjunto de treino, encontram-se **devidamente explicados na Tarefe 2. b.**, e implementados dentro de um ***metanode*** criado (***Metanode test data***).

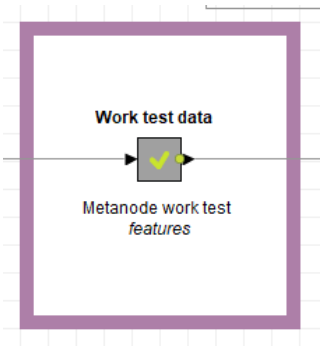


Figura nº114 – ***Metanode*** Implementado para Tratamento de Dados Teste

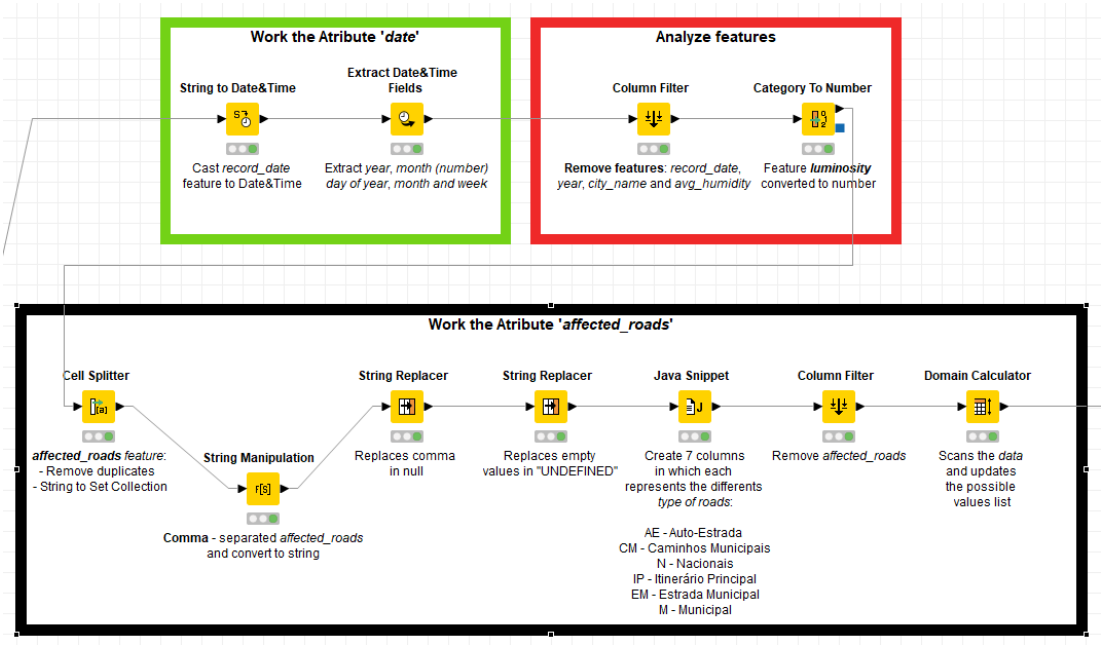


Figura nº115 –*Workfkow* Implementado para Tratamento de Dados de Teste

Para a implementação do nosso modelo, resolvemos recorrer às ***Random Forest***. No nodo ***Random Forest Learner***, e depois de ter feito o tratamento e *tuning* dos dados, resolvemos excluir do nosso modelo as *features avg_humidity*, *luminosity*, *avg_atm_pressure* e *avg_precipitation*. Ao nível do ***Split Criterion***, escolhemos o ***Gini Index***. No que ao número de modelos diz respeito, escolhemos **400 modelos**. De salientar que todos estas configurações não foram escolhidas ao acaso, pois foram estas configurações que extraímos do *tuning* do modelo, de modo a obter um modelo "ótimo" e com o maior valor possível para a *accuracy*.

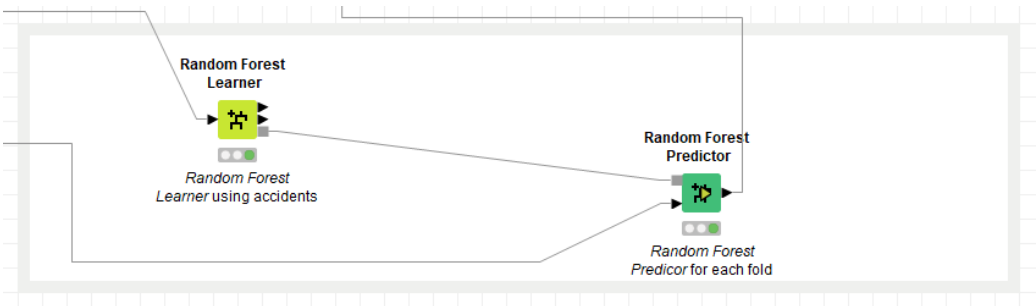


Figura nº116 – Teste do Modelo

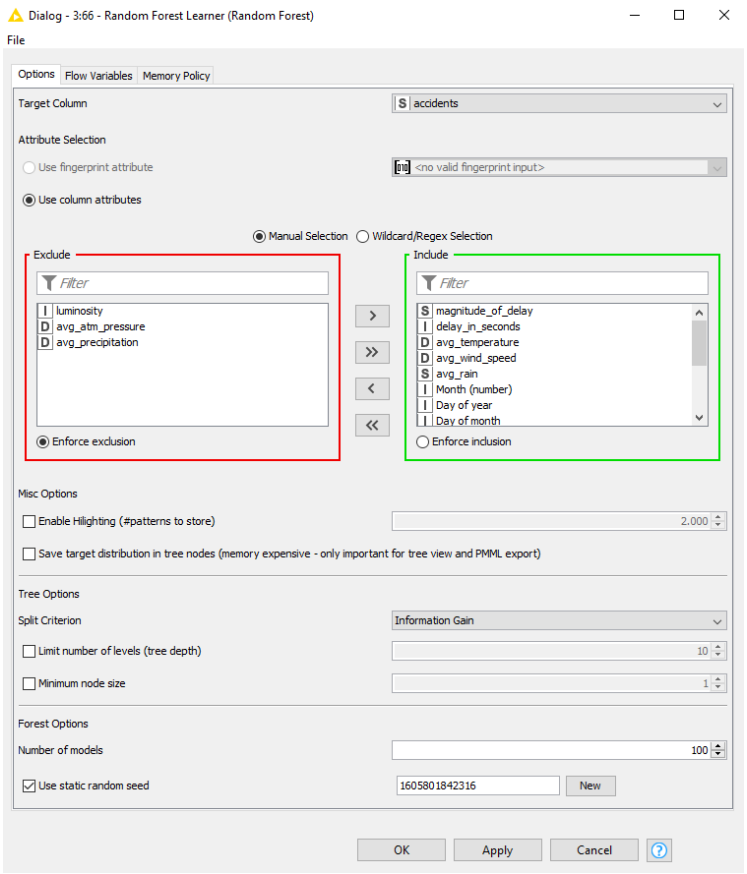


Figura nº117 – Configurações do nodo ***Random Forest Learner***

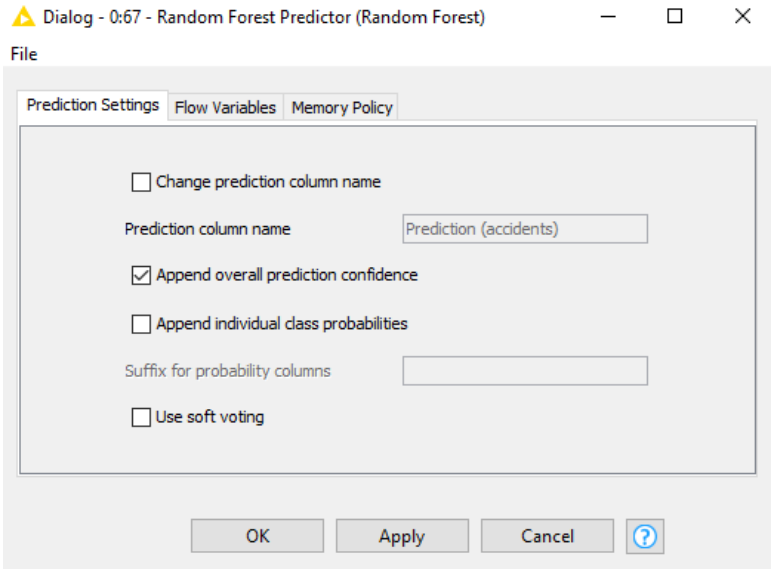


Figura nº118 – Configurações do nodo ***Random Forest Predictor***

Uma vez que um dos objetivos deste trabalho prático passava também por termos uma competição entre os outros grupos de trabalho da UC de Sistemas Baseados em Similaridade

do Perfil de Machine Learning: Fundamentos e Aplicações do MIEI/MEI e da UC Sistemas Baseados em Similaridade do MES da UMinho, foi então necessário utilizarmos e implementarmos alguns nodos no nosso *workflow*, que nos permitissem gerar um ficheiro do tipo *.csv*, de modo conseguirmos submeter o mesmo no *Kaggle* da competição. Na imagem que se segue pudemos observar os nodos implementados.

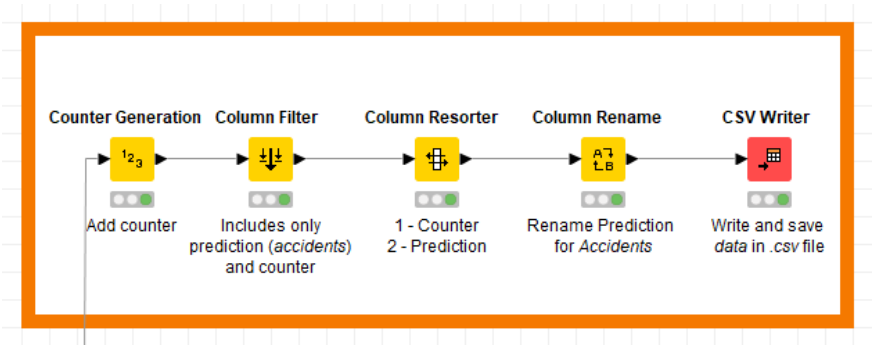


Figura nº119 – Nodos Implementados para Gerar o Ficheiro *.csv*

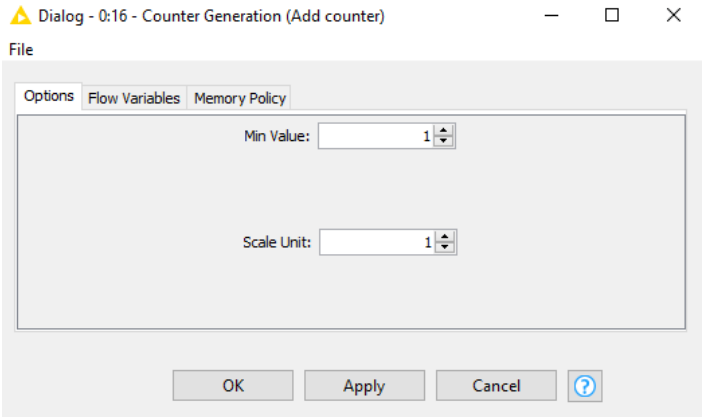


Figura nº120 – Configurações do nodo ***Counter Generation***

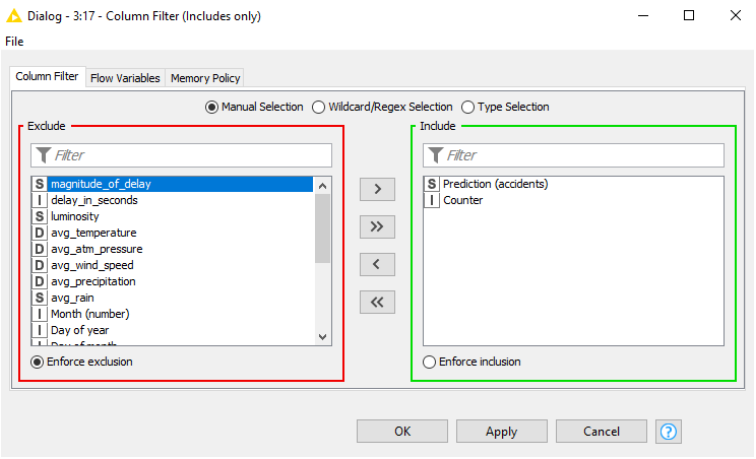


Figura nº121 – Configurações do nodo ***Column Filter***

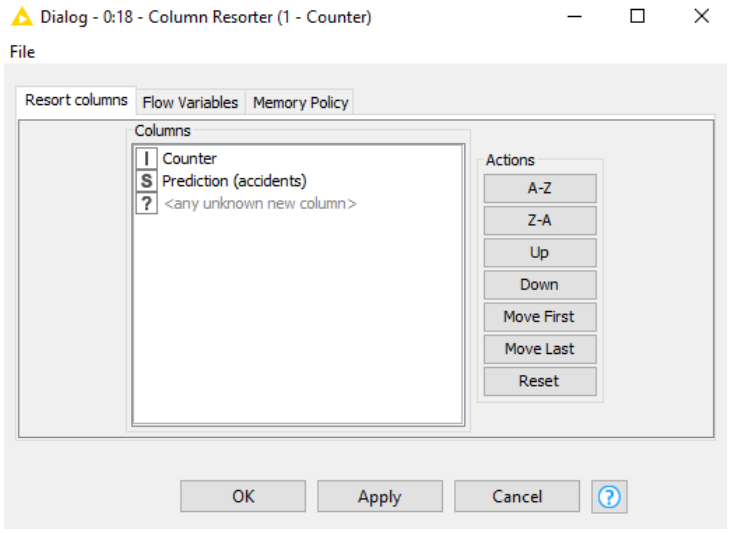


Figura nº122 – Configurações do nodo ***Column Resorter***

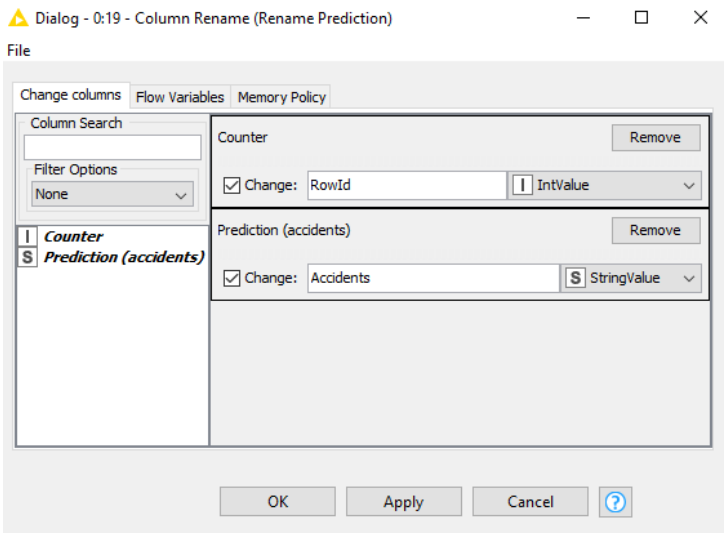


Figura nº123 – Configurações do nodo ***Column Rename***

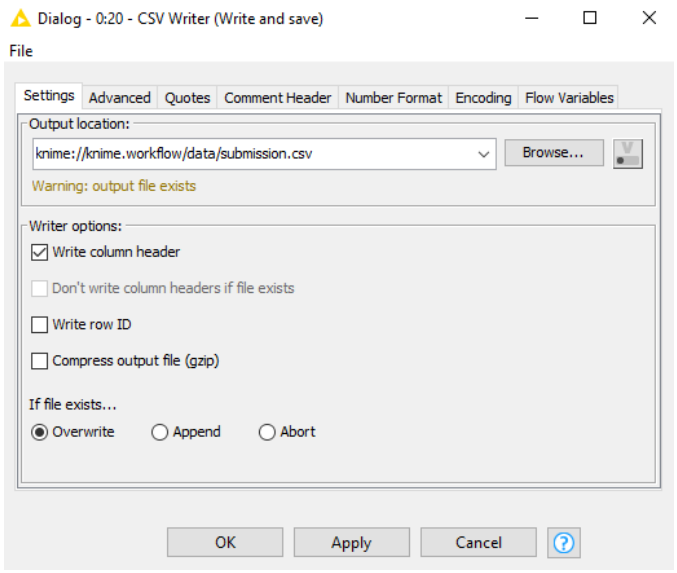


Figura nº124 – Configurações do nodo ***CSV Writter***

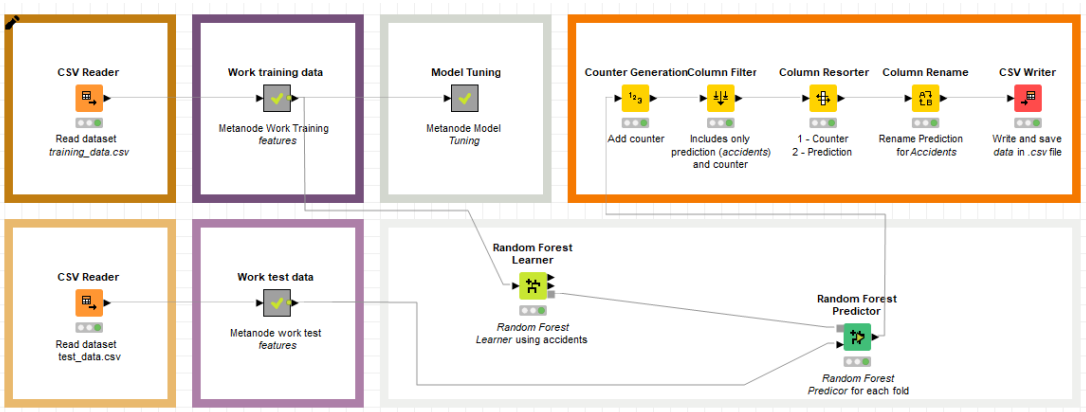


Figura nº125 – *Output* Final do *Workflow* Implementado

f. Evidências e Explicações de Outras Abordagens

Para além da abordagem acima enunciada, e escolhida por nós como "ótima", ao longo da elaboração de todo o trabalho prático, fomos fazendo e implementando várias abordagens, até chegar aquela que nós consideramos a melhor. Deste modo, resolvemos apresentar algumas das outras implementações feitas por nós, uma vez que consideramos importante evidenciar a nossa evolução ao longo deste trabalho, e também, na nossa opinião, estas abordagens foram de extremamente importantes, pois permitiram-nos ir tirando diversas conclusões que sem sombra de dúvida nos ajudaram a encontrar e a perceber como chegar à solução ótima.

1. 1ª Abordagem – *Accuracy* de 0.76373%

Numa primeira tentativa da interpretação do *dataset*, a questão debruçou-se essencialmente sobre qual modelo de treino que nos permitiria obter uma melhor *accurary*, ***Random Forest Learner*** (ao longo das tentativas, evidenciamos que seria o melhor modelo) ou ***Decision Tree Learner***. Para isso, foi necessário compreender os dados e laborá-los. Neste processo, exploramos os dados com o nodo ***Data Explorer***, com o objetivo de a obter medianas, médias, variância e as diversas estatísticas (medidas de dispersão central).

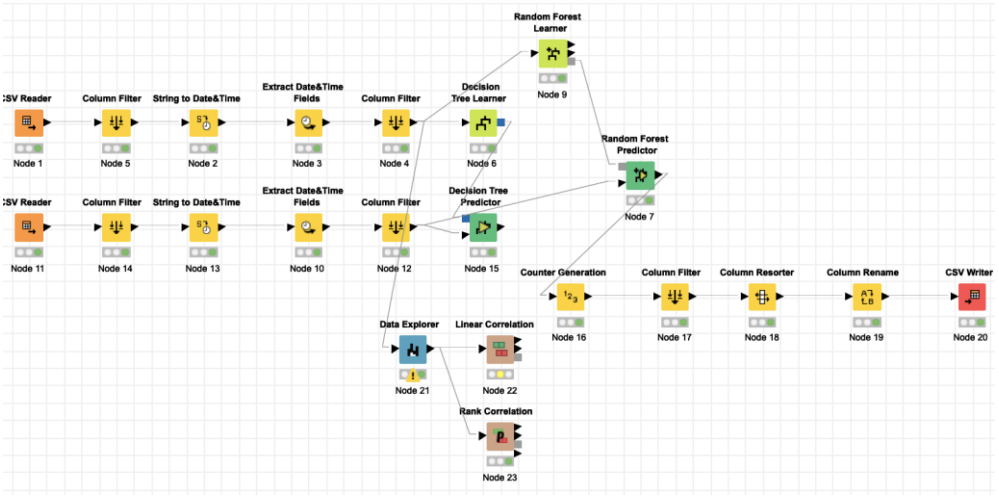


Figura nº126 – *Workflow* Implementado na 1ª Abordagem

2. 5ª Abordagem – *Accuracy* de 0.89010%

Como os incidentes poderiam estar relacionados com as horas de maior pico de trânsito, consideramos extrair dados da *feature* "*record_date*", como mês (número), dia do ano, dia do mês e dia da semana (número), de maneira a obter maior precisão. Para facilitar a análise da *accuracy*, adicionamos uma partição de 70-30 nos dados treinados.

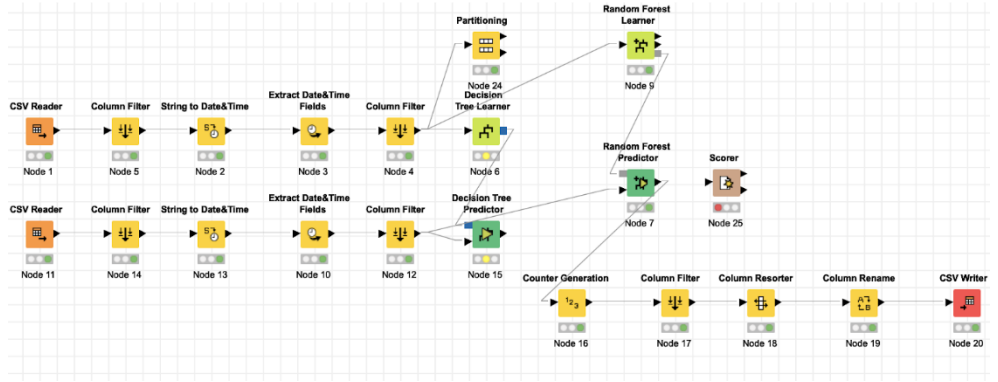


Figura nº127 – *Workflow* Implementado na 5ª Abordagem

3. 6ª à 10ª Abordagem – *Accuracy* de 0.90659%

Ao longo destas abordagens formos fazendo uma análise sequencial das diversas *features*, tais como: **remoção** e **adição**, **tratamento de *missing values*** por cada tipo de dados, **normalização**, ***auto-binners*** e **correlação** entre *features*.

I. 6ª Abordagem – Accuracy de 0.90659%

À semelhança da abordagem feita anteriormente, e não alterando nada no *workflow* até então construído, na 6ª abordagem, decidimos colocar do nodo 9 (***Random Forest Leaner***), e decidimos excluir as seguintes features:

- *city_name*
- *affected_roads*
- *luminosity*
- *avg_atm_pressure*
- *avg_humidity*
- *avg_precipitation*

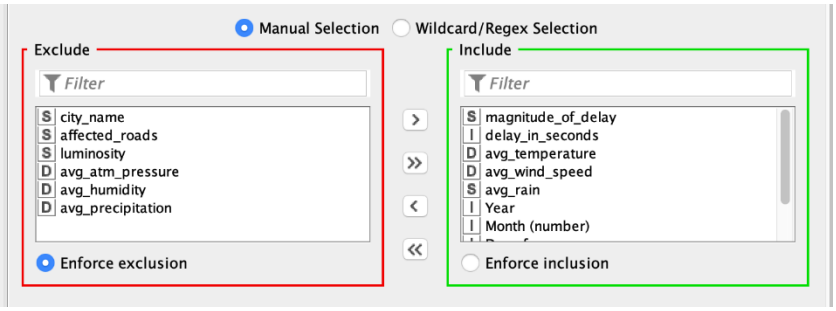


Figura nº128 – Configurações do Nodo ***Random Forest Learner***

II. 7ª Abordagem – Accuracy de 0.90659%

Nesta abordagem resolvemos remover a feature city_name do nodo 4 (***Column Filter***) em vez de excluir do nodo 9 (***Random Forest Leaner***). Contudo, a nossa *accuracy* não se alterou.

III. 8ª Abordagem – Accuracy de 0.90659%

Nesta abordagem, mantivemos tudo à semelhança da 7ª abordagem, alterando apenas a exclusão de uma *feature* (*city_name*) do nodo 12 (***Column Filter***). Contudo, e mais uma vez, não vimos a nossa *accuracy* aumentar.

IV. 9ª Abordagem – Accuracy de 0.90659%

Nesta abordagem, mantivemos tudo à semelhança da 8ª abordagem, alterando apenas a exclusão de algumas *features* (*afected_roads*, *pressure* e *precipitaion*) do nodo 12 (***Column Filter***). Contudo, a nossa *accuracy* não aumentou.

V. 10ª Abordagem – Accuracy de 0.90659%

Na 10ª abordagem, e uma vez que não estávamos a obter os resultados que pretendíamos com as iterações anteriores feitas ao modelo, resolvemos fazer o tratamento dos dados nulos.

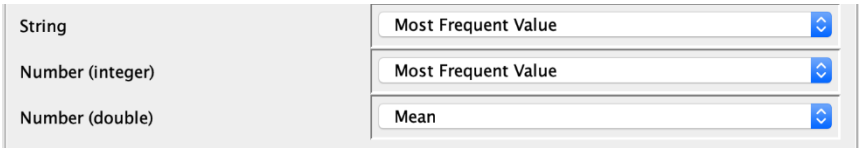


Figura nº129 – Tratamento de ***Missing Values***

Tuning e Análise das melhores *features*

Com a introdução ao *tuning* do modelo e análise das melhores *features*, conseguimos estudar de forma mais pormenorizada as experiências anteriores, com o objetivo de atingir a melhor *accurary* possível. Durante todo este processo e avaliação de *features* e *tuning*, conseguimos que a nossa *accurary* se encontrasse compreendida entre os 90% e 95%.

Melhor *Performance*

Conseguimos adquirir a melhor *performance* do modelo, através de várias experiências com a *feature* ***affected_roads***, uma vez que esta tinha dados demasiado compactos e pouco claros. Numa primeira abordagem desagregamos a *String* e efetuamos uma transformação guardando os dados numa coleção. Posto isto, e com o auxílio do nodo ***Java Snippet*** geramos 6 colunas baseadas no tipo de estrada, em que cada uma guardava um *boleano* baseado na informação retirada da coleção. De acordo com a tabela seguinte, conseguimos visualizar os tipos de estradas armazenados.

affected_roads	AE	CM	N	IP	EM	M
N101	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE
CM1348, A11, N103	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE

Tabela nº1 – Extração de Informação Efetuada da *Feature* ***affected_roads*** (Tipo de Estrada)

Após observar a *accurary* final do modelo desenvolvido, conseguimos perceber que o mesmo ainda poderia melhorar, se fosse feita outra abordagem. Esta passou por manter as 6 colunas, mas ao invés de armazenar variáveis *boleanas* em cada coluna, passamos a armazenar o nome das estradas baseadas no seu tipo. De acordo com a tabela seguinte, conseguimos visualizar os nomes das estradas armazenados.

affected_roads	AE	CM	N	IP	EM	M
N101	N/A	N/A	101	N/A	N/A	N/A
CM1348, A11, N103	11	1248	103	N/A	N/A	N/A

Tabela nº2 – Extração de Informação Efetuada da *Feature* ***affected_roads*** (Nome da Estrada)

Deste modo e para concluir, a análise das estradas passou a ser mais clara e detalhada, e assim alcançamos uma *accurary* de aproximadamente 95%.

Conclusão.

Com a elaboração deste projeto/trabalho prático, cujo objetivo principal recaia sobre a construção de dois modelos de previsão baseado em *machine learning*, de modo a obtermos a melhor *accuracy* possível, não só para a competição no *Kaggle*, mas também no d*ataset* escolhido por nós.

Durante o desenvolvimento dos modelos, foi crucial a análise aos resultados que íamos obtendo, uma vez que no processo de desenvolvimento de modelos de *machine learning*, é importante, não só, saber analisar os dados e obter conclusões sobre estes, mas também, encontrar formas de moldar os dados de modo a obtermos os resultados esperados.

Com a realização deste trabalho prático e com o fim deste relatório, esperamos ter atingido todos os pontos que nos foram propostos e, conseguido explicar todo o desenvolvimento nas nossas etapas para que, fosse claro demonstrar as nossas análises e explorações realizadas em ambos os *datasets*.