AIRTICKETS – FINAL PROJECT

Tomancová Nikola

OBSAH

POPIS DATASETU A ZKOUMANÉ OBLASTI

PŘÍPRAVA DAT A ANALÝZA V POWER BI

DATA MINING – ANALÝZA V CLEVERMINERU

POPIS ŘEŠENÉ OBLASTI

- Analýza prodejů letenek
- Sledované období: **17. dubna 2022 až 5. května 2022**
- Sledovaná oblast: **USA (vnitrostátní lety)**
- Zkoumání faktorů ovlivňujících ceny letenek
- Pozorování rozdílů např. pro různé cestovní třídy, různé dny v týdnu, různé destinace, meteorologické podmínky…
- Hledání souvislostí mezi atributy

DATASET
FLIGHT PRICES

- Zdroj: Flight Prices USA 2022
- CSV soubor
- Primární dataset
- Údaje o prodávaných letenkách na americké vnitrostátní lety ve sledovaném období
- Ceny letenek, časy odletů a příletů, vzdálenosti, třída apod.
- Původní dataset následně upraven v Pythonu

DATASET AIRPORT CODES

- Zdroj: Airport Codes USA
- Soubor .xls
- Slouží jako zdroj doplňujících informací
- Název letiště
- Město, v němž se letiště nachází
- Původní soubor zjednodušen do výsledného csv souboru

DATASET WEATHER

- Zdroj: The Meteostat Python library
- Python knihovna meteostat
- Historická data o počasí
- Údaje zjištěné na základě geografické polohy letiště
- Pouze pro počáteční a cílové destinace, nikoliv pro destinace přestupní
- Následná kategorizace proměnných tavg (průměrná denní teplota vzduchu v °C), prcp (celkový denní úhrn srážek v mm) a wspd (průměrná rychlost větru v km/h)

WEEKDAY TABLE

- Zdroj: Python (knihovna datetime)
- Slouží jako zdroj doplňujících informací
- Den v týdnu (01 Mon až 07 Sun)

CÍLE ANALÝZY I

- Cílem této práce je analýza letenek na vybrané trasy od různých leteckých společností pro různé hodnoty dalších atributů.
- Data budou zkoumána především z finanční stránky, většina analytických otázek se proto bude týkat především cen letenek, tzn. budou zkoumány hlavně vztahy atributu totalFare_cat s ostatními dostupnými atributy.

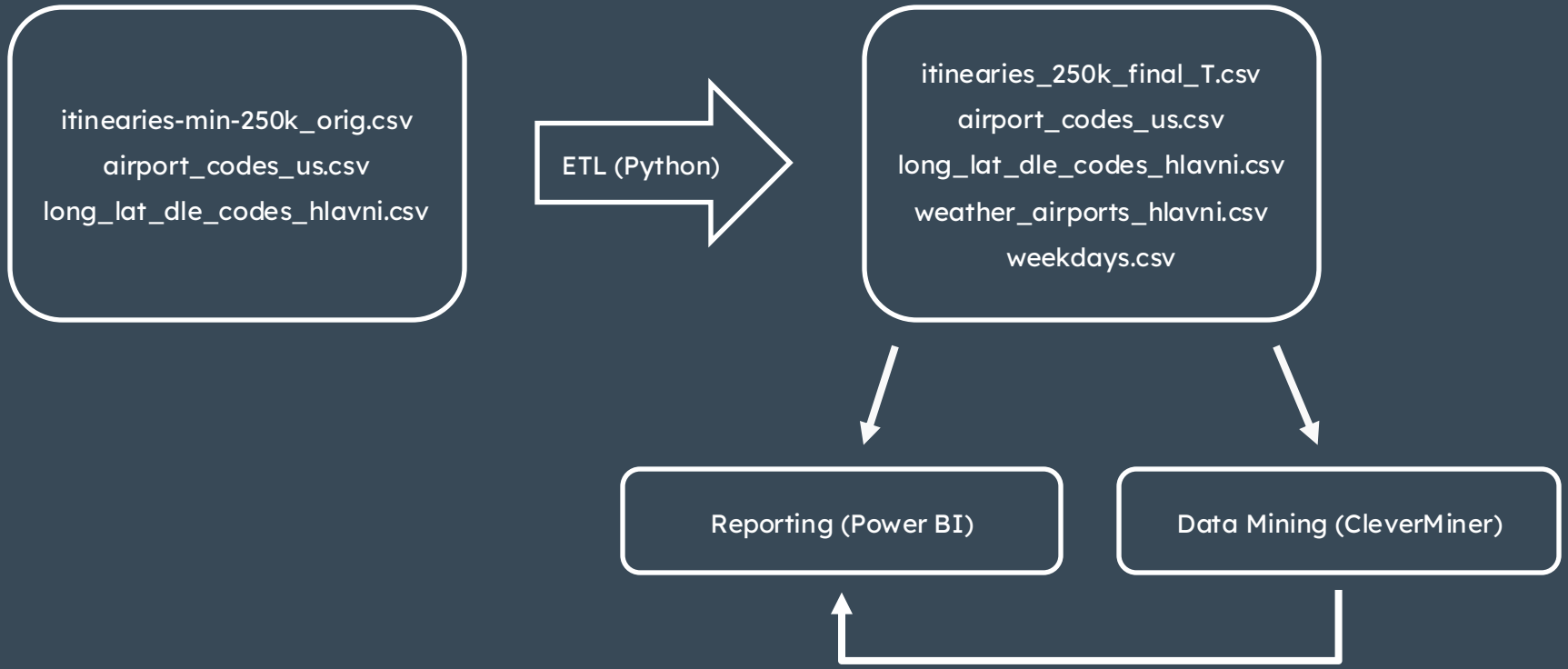
CÍLE ANALÝZY II

Účelem této práce je pomoci s průzkumem pro začínající leteckou společnost, která zvažuje, jak vysoké ceny letenek by měla stanovit či na jaké trasy by bylo vhodné se soustředit.

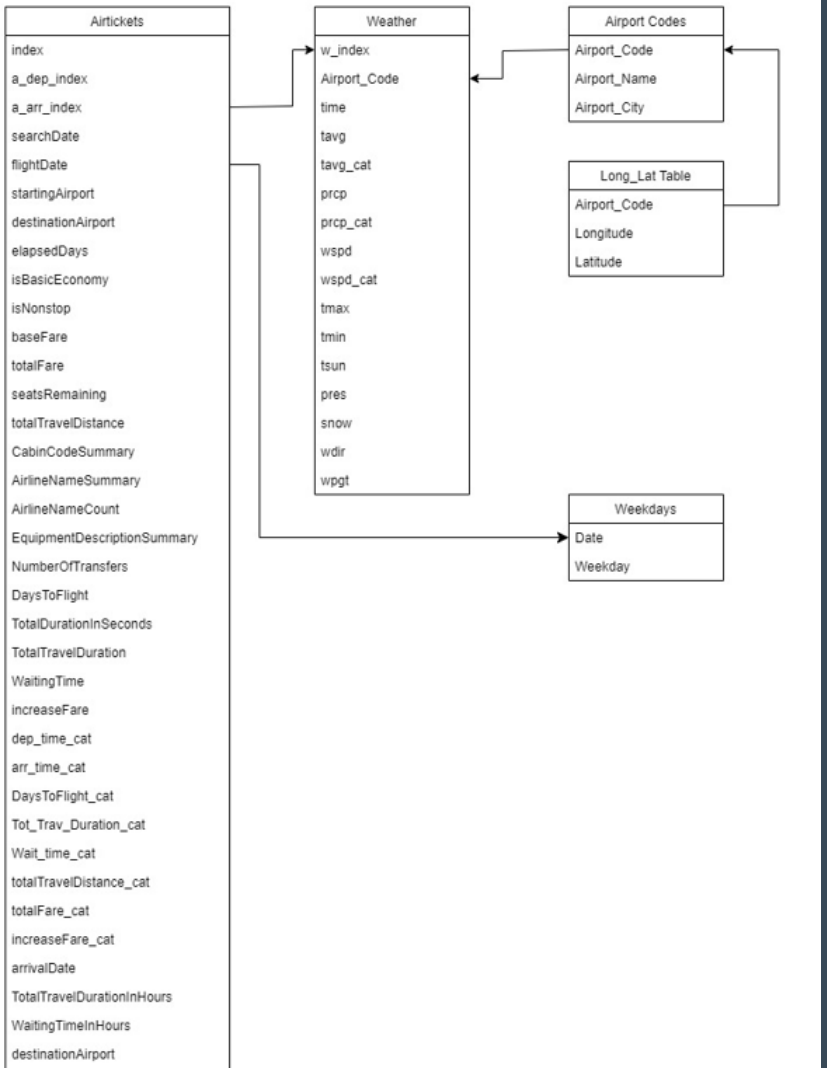
To tedy mimo jiné znamená:

- Identifikaci faktorů ovlivňujících cenu letenky
- Hledání zajímavých souvislostí v datech
- Návrh využití poznatků v praxi

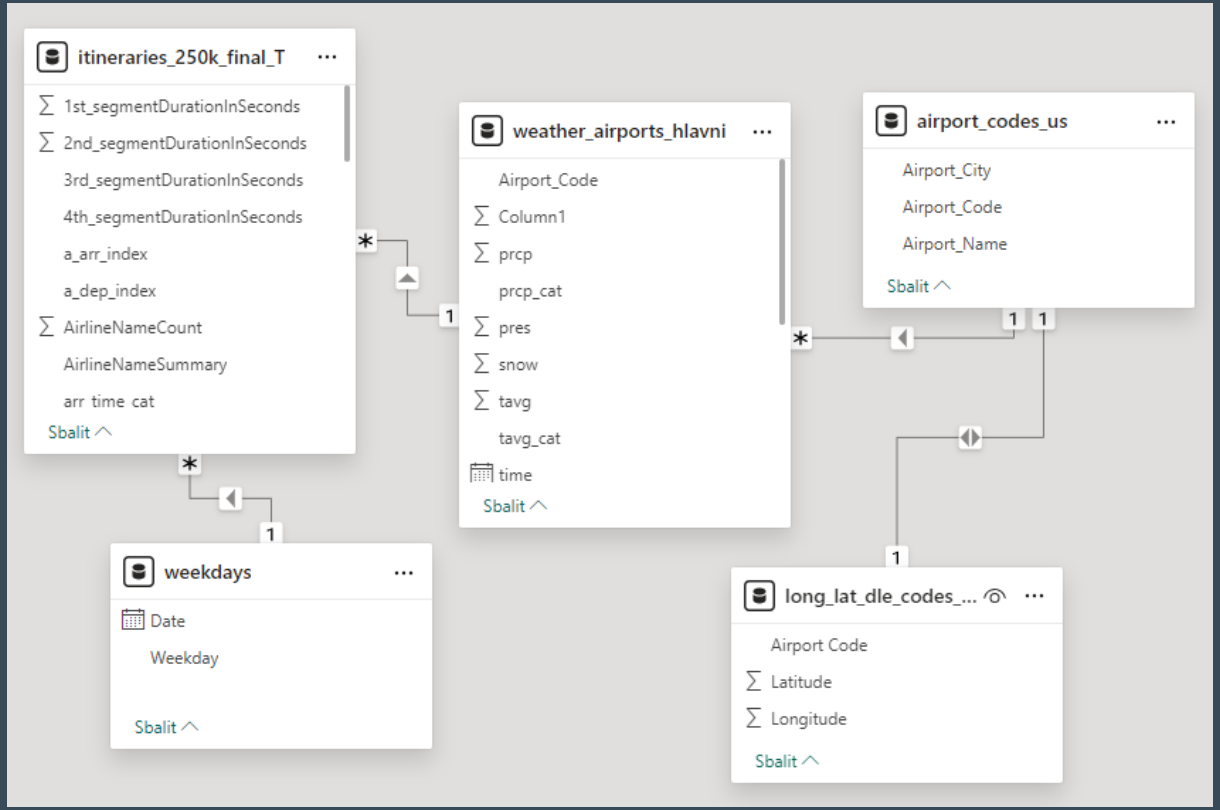
ARCHITEKTURA ŘEŠENÍ



DATOVÝ MODEL



DATOVÝ MODEL V POWER BI



PŘÍPRAVA DAT

Datové pumpy (ETL) – komentované kódy z Pythonu

01

WEATHER TABLE

- Pomocí Python knihovny meteostat získány údaje o počasí pro dané období na základě zeměpisné délky a šířky letišť (pouze počáteční a cílové destinace).
- Dále byly využity pouze některé z nově získaných atributů.

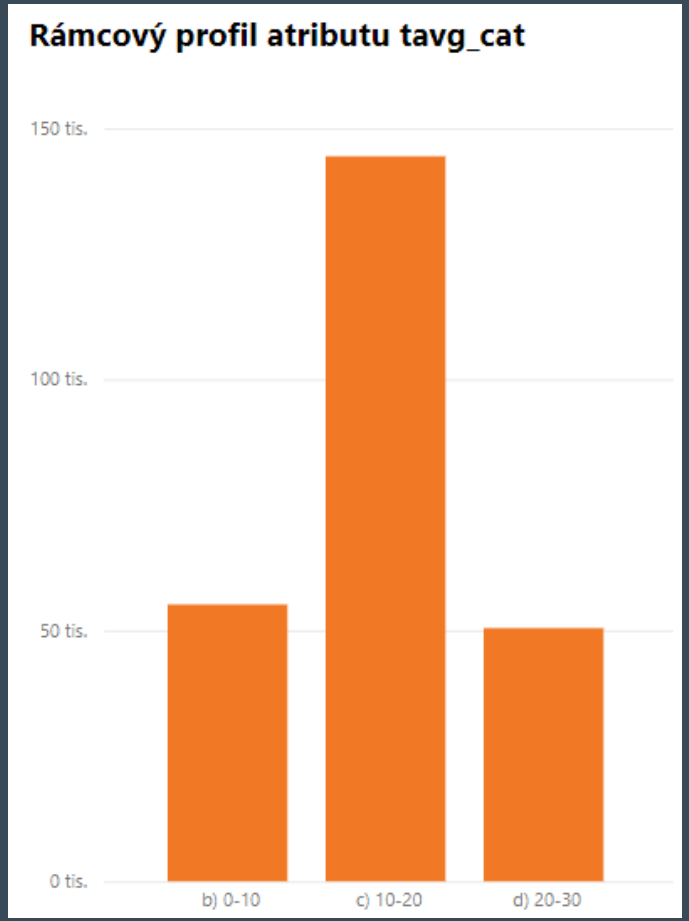
```python
# Set time period
start = datetime(2022, 4, 10)
end = datetime(2022, 5, 12)

df = pd.read_csv('long_lat_dle_codes_hlavni.csv', sep=";")
codes = df['Airport Code'].tolist()
lat = df['Latitude'].tolist()
long = df['Longitude'].tolist()
weather_table = pd.DataFrame()

for i in df.index:
    # Create Point
    location = Point(lat[i],long[i])
    # Get daily data
    data = Daily(location, start, end).fetch()
    data['Airport_Code']=codes[i]
    weather_table = pd.concat([weather_table, data])
```

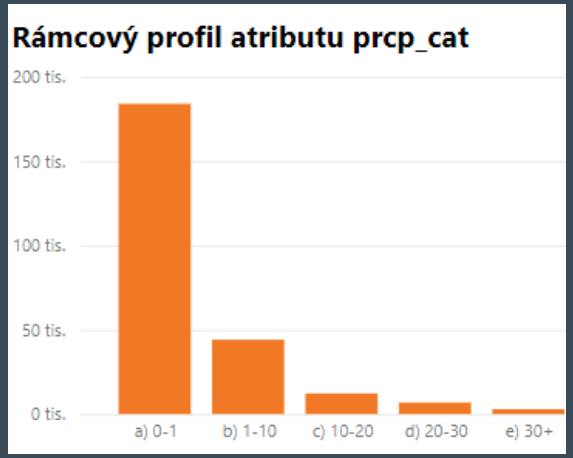
- Kategorizace atributů **tavg** (průměrná denní teplota vzduchu ve °C), **prcp** (celkový denní úhrn srážek v mm) a **wspd** (průměrná denní rychlost větru v km/h)
- Vytvoření unikátního ID (sloupec w_index, formát „kód letiště & datum“)

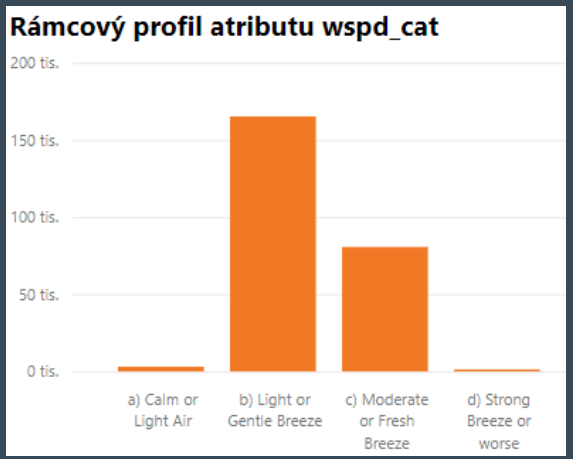
```
weather_table['tavg_cat'] = pd.cut(weather_table['tavg'],
                                   bins = [-float("inf"),0, 10,20,30, float("inf")],
                                   labels = ['a) <0', 'b) 0-10', 'c) 10-20',
                                             'd) 20-30', 'e) 30+'])
weather_table['prcp_cat'] = pd.cut(weather_table['prcp'],
                                   bins = [-float("inf"), 1,10,20,30, float("inf")],
                                   labels = ['a) 0-1', 'b) 1-10', 'c) 10-20',
                                             'd) 20-30', 'e) 30+'])
weather_table['wspd_cat'] = pd.cut(weather_table['wspd'],
                                   bins = [-float("inf"),5,19,38, float("inf")],
                                   labels = ['a) Calm or Light Air',
                                             'b) Light or Gentle Breeze',
                                             'c) Moderate or Fresh Breeze',
                                             'd) Strong Breeze or worse'])
df.rename(columns = {'Unnamed: 0' : 'w_index'}, inplace = True)
df['w_index'] = df['Airport_Code'] + ' ' + df['time']
```

	w_index	time	tavg	tmin	tmax	prcp	snow	wdir	wspd	wpgt	pres	tsun	Airport_Code	tavg_cat	prcp_cat	wspd_cat
0	ATL 2022-04-10	2022-04-10	12.0	0.6	23.3	0.0	0.0	303.0	5.2	NaN	1018.5	NaN	ATL	c) 10-20	a) 0-1	b) Light or Gentle Breeze
1	ATL 2022-04-11	2022-04-11	14.4	6.7	21.1	0.0	0.0	308.0	4.7	NaN	1019.6	NaN	ATL	c) 10-20	a) 0-1	a) Calm or Light Air
2	ATL 2022-04-12	2022-04-12	19.4	11.7	27.2	0.0	0.0	265.0	7.6	NaN	1019.4	NaN	ATL	c) 10-20	a) 0-1	b) Light or Gentle Breeze
3	ATL 2022-04-13	2022-04-13	21.2	16.7	25.0	0.0	0.0	173.0	12.2	NaN	1015.8	NaN	ATL	d) 20-30	a) 0-1	b) Light or Gentle Breeze
4	ATL 2022-04-14	2022-04-14	19.7	15.6	23.9	1.0	0.0	283.0	12.2	NaN	1014.9	NaN	ATL	c) 10-20	a) 0-1	b) Light or Gentle Breeze
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
523	SFO 2022-05-08	2022-05-08	12.6	10.0	16.1	0.0	NaN	293.0	31.7	NaN	1016.7	NaN	SFO	c) 10-20	a) 0-1	c) Moderate or Fresh Breeze
524	SFO 2022-05-09	2022-05-09	12.3	8.9	16.1	0.0	NaN	278.0	18.4	NaN	1018.3	NaN	SFO	c) 10-20	a) 0-1	b) Light or Gentle Breeze
525	SFO 2022-05-10	2022-05-10	12.1	9.4	15.6	0.0	NaN	285.0	27.0	NaN	1021.9	NaN	SFO	c) 10-20	a) 0-1	c) Moderate or Fresh Breeze
526	SFO 2022-05-11	2022-05-11	12.3	8.3	16.7	0.0	NaN	288.0	26.6	NaN	1024.3	NaN	SFO	c) 10-20	a) 0-1	c) Moderate or Fresh Breeze
527	SFO 2022-05-12	2022-05-12	12.9	10.0	17.8	0.0	NaN	292.0	29.5	NaN	1026.4	NaN	SFO	c) 10-20	a) 0-1	c) Moderate or Fresh Breeze

528 rows × 16 columns


Rámcový profil atributu tavg_cat
150 tis.
100 tis.
50 tis.
0 tis.
b) 0-10
c) 10-20
d) 20-30

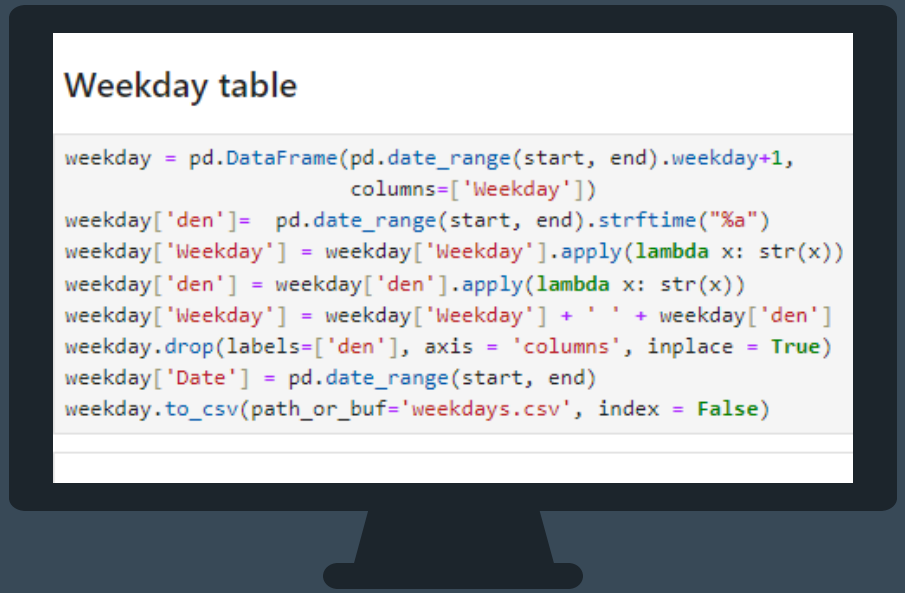

Rámcový profil atributu prcp_cat
200 tis.
150 tis.
100 tis.
50 tis.
0 tis.
a) 0-1
b) 1-10
c) 10-20
d) 20-30
e) 30+


Rámcový profil atributu wspd_cat
200 tis.
150 tis.
100 tis.
50 tis.
0 tis.
a) Calm or Light Air
b) Light or Gentle Breeze
c) Moderate or Fresh Breeze
d) Strong Breeze or worse

02

WEEKDAY TABLE

- Tabulka obsahující 2 sloupce:
  - Date (datum od 17. 4. do 5. 5. 2022)
  - Weekday (odpovídající den v týdnu ve formátu „pořadí dne v týdnu & ang. zkratka názvu dne v týdnu“

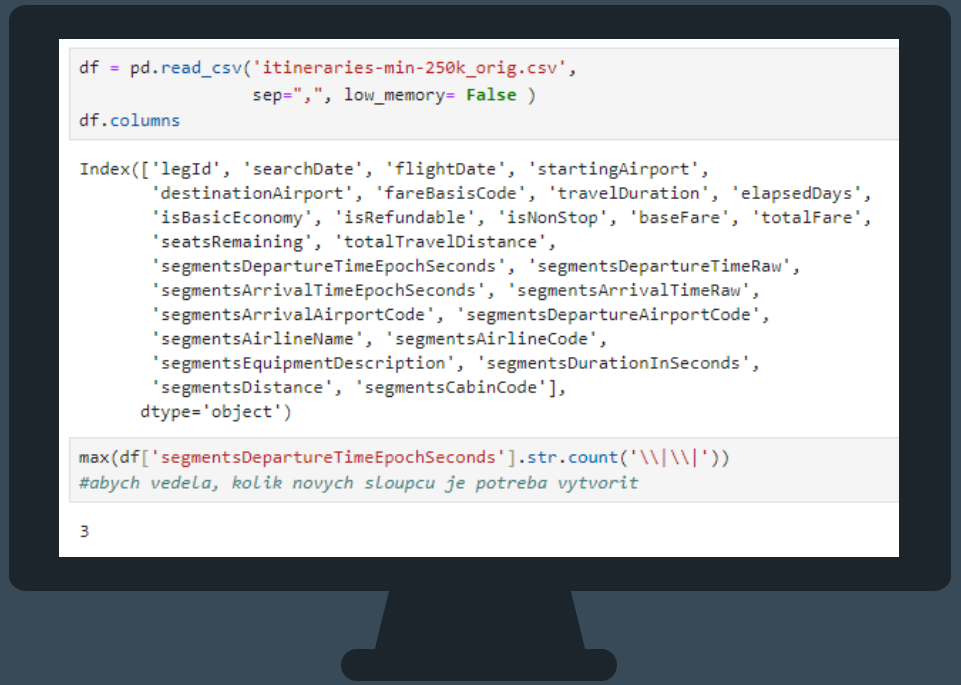


	Weekday	Date		Weekday	Date		Weekday	Date
0	7 Sun	2022-04-10	11	4 Thu	2022-04-21	22	1 Mon	2022-05-02
1	1 Mon	2022-04-11	12	5 Fri	2022-04-22	23	2 Tue	2022-05-03
2	2 Tue	2022-04-12	13	6 Sat	2022-04-23	24	3 Wed	2022-05-04
3	3 Wed	2022-04-13	14	7 Sun	2022-04-24	25	4 Thu	2022-05-05
4	4 Thu	2022-04-14	15	1 Mon	2022-04-25	26	5 Fri	2022-05-06
5	5 Fri	2022-04-15	16	2 Tue	2022-04-26	27	6 Sat	2022-05-07
6	6 Sat	2022-04-16	17	3 Wed	2022-04-27	28	7 Sun	2022-05-08
7	7 Sun	2022-04-17	18	4 Thu	2022-04-28	29	1 Mon	2022-05-09
8	1 Mon	2022-04-18	19	5 Fri	2022-04-29	30	2 Tue	2022-05-10
9	2 Tue	2022-04-19	20	6 Sat	2022-04-30	31	3 Wed	2022-05-11
10	3 Wed	2022-04-20	21	7 Sun	2022-05-01	32	4 Thu	2022-05-12

03

AIR TICKETS TABLE

- Původně 27 atributů, 250 000 řádků
- Atributy s názvem „segment…“ obsahují informace postupně pro všechny dílčí lety v rámci cesty (navzájem oddělené pomocí „||“)
- Všechny lety mají max. 3 přestupy



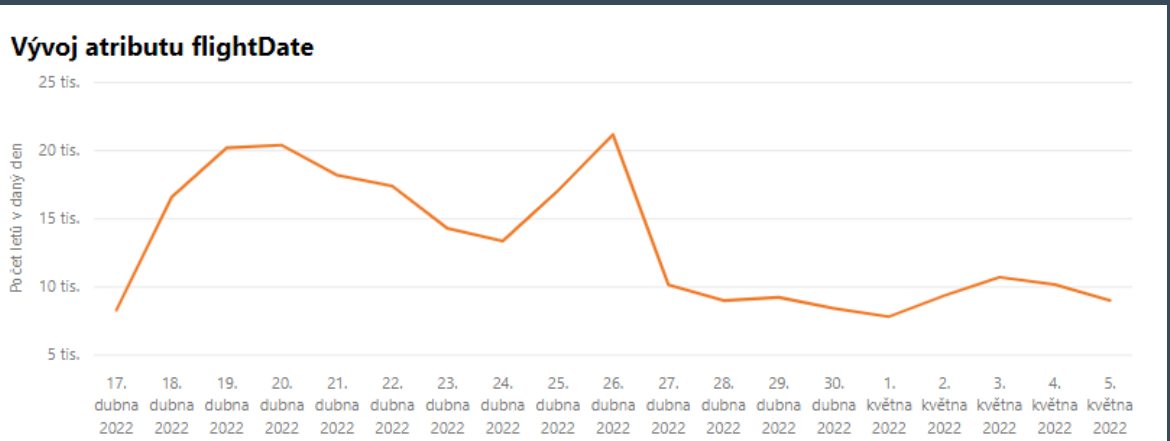
```
df = pd.read_csv('itineraries-min-250k_orig.csv',
                 sep=",", low_memory= False )
df.columns
```

```
Index(['legId', 'searchDate', 'flightDate', 'startingAirport',
       'destinationAirport', 'fareBasisCode', 'travelDuration', 'elapsedDays',
       'isBasicEconomy', 'isRefundable', 'isNonStop', 'baseFare', 'totalFare',
       'seatsRemaining', 'totalTravelDistance',
       'segmentsDepartureTimeEpochSeconds', 'segmentsDepartureTimeRaw',
       'segmentsArrivalTimeEpochSeconds', 'segmentsArrivalTimeRaw',
       'segmentsArrivalAirportCode', 'segmentsDepartureAirportCode',
       'segmentsAirlineName', 'segmentsAirlineCode',
       'segmentsEquipmentDescription', 'segmentsDurationInSeconds',
       'segmentsDistance', 'segmentsCabinCode'],
      dtype='object')
```

```
max(df['segmentsDepartureTimeEpochSeconds'].str.count('\\|\\|'))
#abych vedela, kolik novych sloupcu je potreba vytvorit
```

```
3
```

searchDate	Počet
16. dubna 2022	87657
17. dubna 2022	162343
Celkem	**250000**

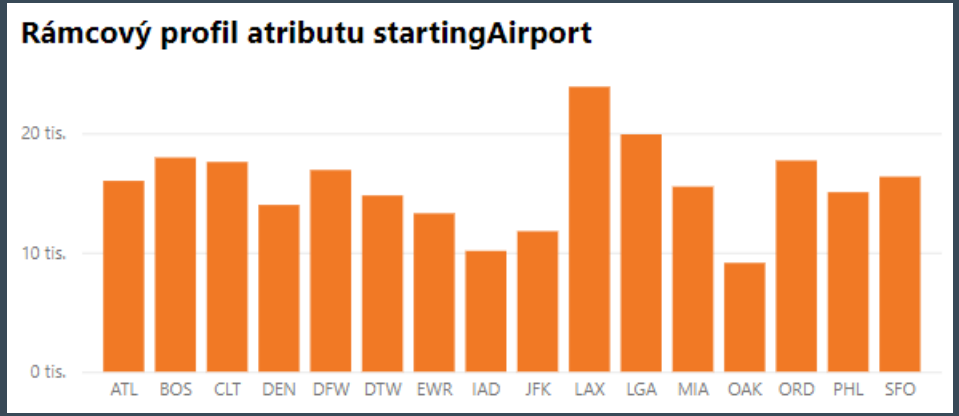


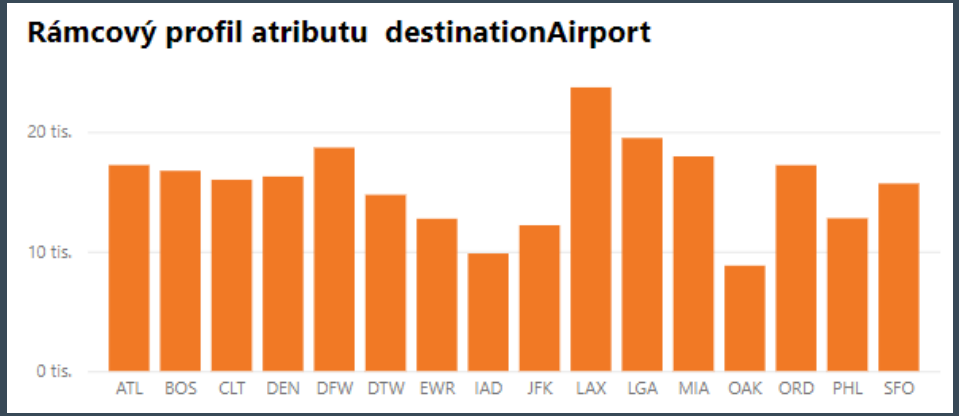
Atribut **searchDate** jako datum vyhledání daného letu na webových stránkách expedia.com

Atribut **flightDate** jako datum odletu z počáteční destinace

Atribut **startingAirport** (**destinationAirport**) jako kód výchozího (cílového) letiště

Airport_Code	Airport_City
ATL	Atlanta
BOS	Boston
CLT	Charlotte
DEN	Denver
DFW	Dallas
DTW	Detroit
EWR	Newark
IAD	Dulles, DC
JFK	New York
LAX	Los Angeles
LGA	New York
MIA	Miami
OAK	Oakland
ORD	Chicago
PHL	Philadelphia
SFO	San Francisco





Atribut **elapsedDays** jako počet dnů oplynulých mezi odletem z počáteční destinace a příletem do cílové destinace

- Např. noční lety přes půlnoc mají hodnotu elapsedDays rovnou jedné.
- Cesty, které začínají i končí ve stejný den, mají hodnotu elapsedDays rovnou nule.

Booleovský atribut **isNonStop** určující, zda je daný let přímý (isNonStop roven jedné), nebo s přestupem (isNonStop roven nule)

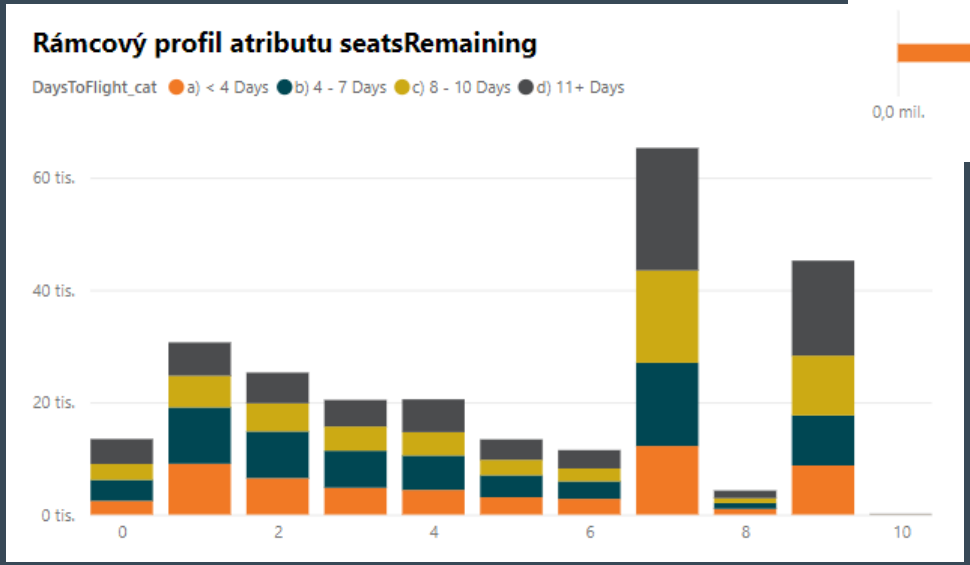
isNonStop	Počet
False	182850
True	67150
Celkem	**250000**

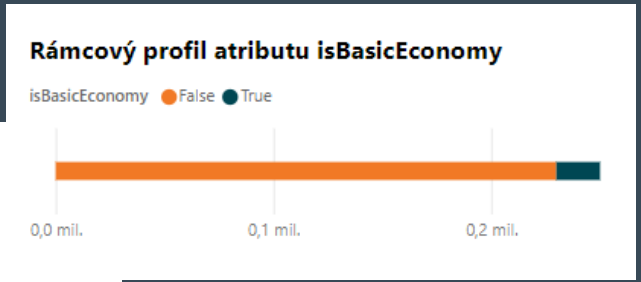
elapsedDays	Počet
0	214192
1	35808
Celkem	**250000**

Booleovský atribut **isBasicEconomy** určujíc,í zda je daná letenka v kategorii basic economy

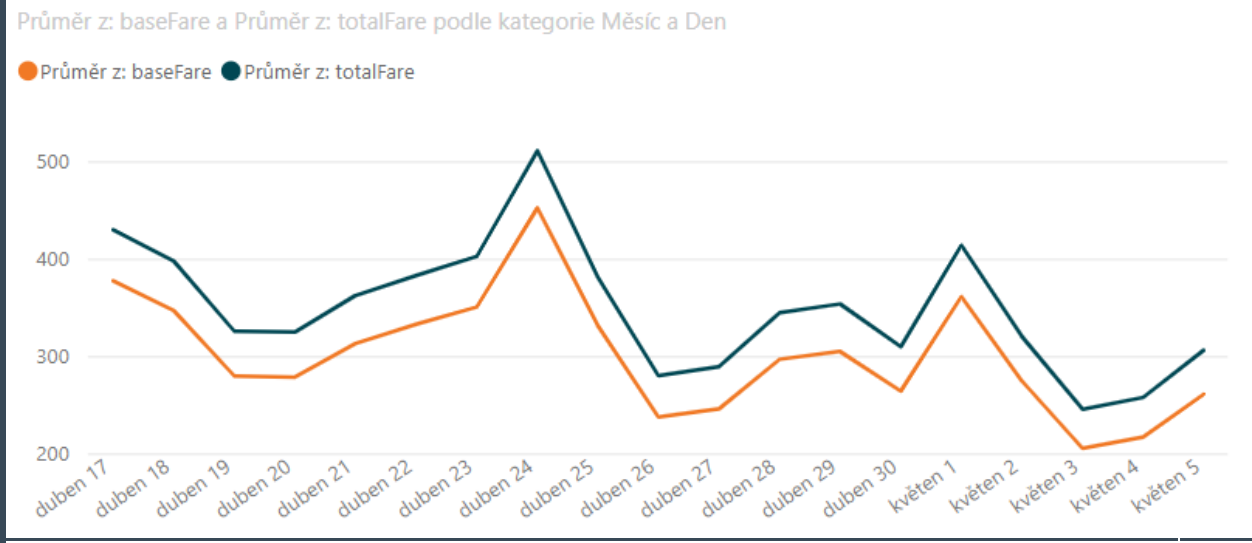
- Obvykle to znamená letenku do třídy Economy bez možnosti volby sedadla, s příručním zavazadlem zdarma a dalším zavazadlem za příplatek, bez možnosti pozdější změny parametrů letenky a s účtováním poplatků za stornování jejího nákupu

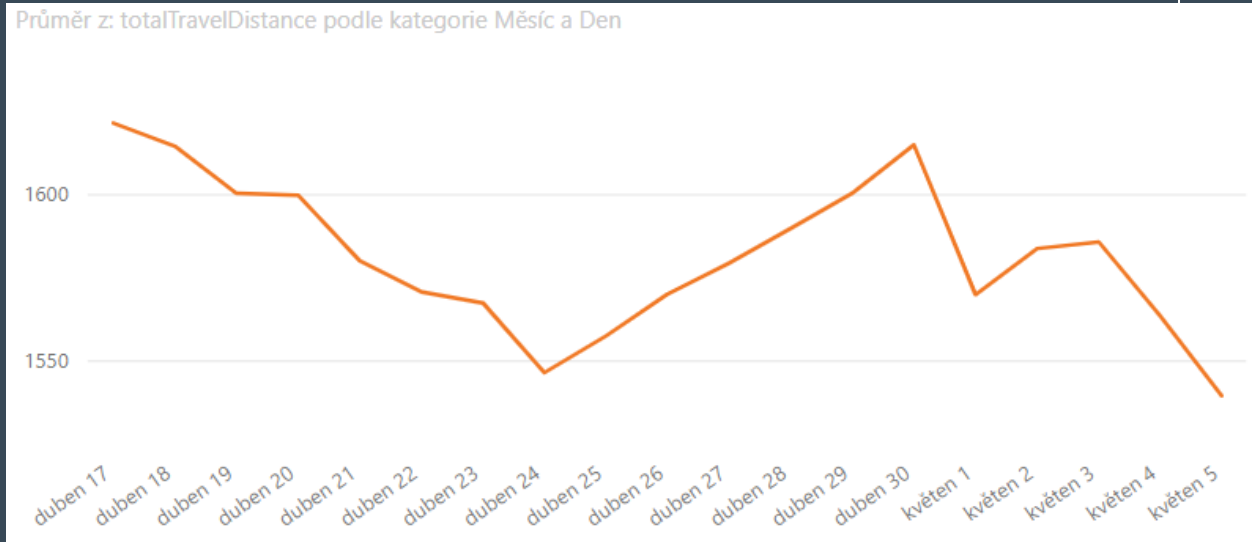
Atribut **seatsRemaining** jako počet zbývajících míst v daném letu, které odpovídají zadaným parametrům


Rámcový profil atributu seatsRemaining
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
60 tis.
40 tis.
20 tis.
0 tis.
0
2
4
6
8
10


Rámcový profil atributu isBasicEconomy
isBasicEconomy
False
True
0,0 mil.
0,1 mil.
0,2 mil.

Atribut **baseFare a totalFare a totalTravelDistance**





- V rámci úprav datového souboru došlo nejprve k rozdělení atributů „segment…“ do 4 sloupců („1st_segment…“, … , „4th_segment…“)

New columns

```
df[['1st_segmentDepartureTimeEpochSeconds','2nd_segmentDepartureTimeEpochSeconds','3rd_segmentDepartureTimeEpochSeconds',
    '4th_segmentDepartureTimeEpochSeconds']]=df['segmentsDepartureTimeEpochSeconds'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentDepartureTimeRaw','2nd_segmentDepartureTimeRaw','3rd_segmentDepartureTimeRaw',
    '4th_segmentDepartureTimeRaw']]=df['segmentsDepartureTimeRaw'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentArrivalTimeEpochSeconds','2nd_segmentArrivalTimeEpochSeconds','3rd_segmentArrivalTimeEpochSeconds',
    '4th_segmentArrivalTimeEpochSeconds']]=df['segmentsArrivalTimeEpochSeconds'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentArrivalTimeRaw','2nd_segmentArrivalTimeRaw','3rd_segmentArrivalTimeRaw',
    '4th_segmentArrivalTimeRaw']]=df['segmentsArrivalTimeRaw'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentArrivalAirportCode','2nd_segmentArrivalAirportCode','3rd_segmentArrivalAirportCode',
    '4th_segmentArrivalAirportCode']]=df['segmentsArrivalAirportCode'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentDepartureAirportCode','2nd_segmentDepartureAirportCode','3rd_segmentDepartureAirportCode',
    '4th_segmentDepartureAirportCode']]=df['segmentsDepartureAirportCode'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentAirlineName','2nd_segmentAirlineName','3rd_segmentAirlineName',
    '4th_segmentAirlineName']]=df['segmentsAirlineName'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentEquipmentDescription','2nd_segmentEquipmentDescription','3rd_segmentEquipmentDescription',
    '4th_segmentEquipmentDescription']]=df['segmentsEquipmentDescription'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentDurationInSeconds','2nd_segmentDurationInSeconds','3rd_segmentDurationInSeconds',
    '4th_segmentDurationInSeconds']]=df['segmentsDurationInSeconds'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentDistance','2nd_segmentDistance','3rd_segmentDistance',
    '4th_segmentDistance']]=df['segmentsDistance'].str.split('\\|\\|', expand = True).fillna('')
df[['1st_segmentCabinCode','2nd_segmentCabinCode','3rd_segmentCabinCode',
    '4th_segmentCabinCode']]=df['segmentsCabinCode'].str.split('\\|\\|', expand = True).fillna('')
```

- Následně bylo odvozeno ještě několik dalších nových atributů.
- Atribut **CabinCodeSummary** jako shrnující atribut obsahující unikátní hodnoty všech cestovních tříd, které byly během daného letu využity (Economy/Coach, Premium Economy/Coach, Business Class, First Class)

New column CabinCodeSummary

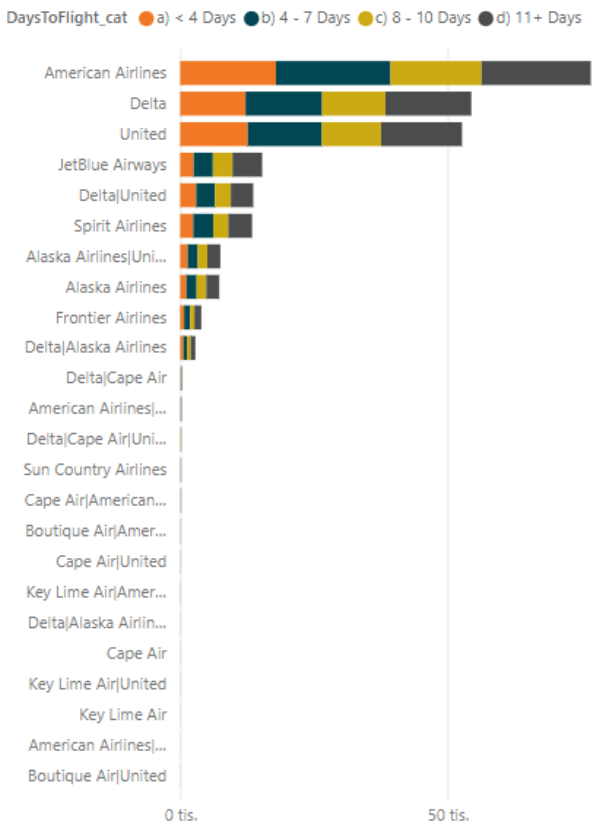
```
def remove_empty(set_value):
    return set(filter(None, set_value))

ccs = df.loc[:,['1st_segmentCabinCode', '2nd_segmentCabinCode',
                '3rd_segmentCabinCode','4th_segmentCabinCode'
               ]].apply(remove_empty, axis=1)

def join_items(set_value):
    return '|'.join(set_value)

df['CabinCodeSummary'] = ccs.apply(join_items)
#kontrola:   df.loc[249911, 'CabinCodeSummary']
```

- Atribut **AirlineNameSummary** jako shrnující atribut obsahující množinu Aerolinek, které byly během daného letu využity
- Atribut **AirlineNameCount** jako počet využitých aerolinek během daného letu (tzn. odvozen jako počet prvků množiny v atributu AirlineNameSummary)

New columns AirlineNameSummary a AirlineNameCount

```
def remove_empty(set_value):
    return set(filter(lambda x: x != "" and not pd.isna(x), set_value))

ccs = df.loc[:,['1st_segmentAirlineName', '2nd_segmentAirlineName',
                '3rd_segmentAirlineName', '4th_segmentAirlineName'
               ]].apply(remove_empty, axis=1)

def join_items(set_value):
    return '|'.join(set_value)

df['AirlineNameSummary'] = ccs.apply(join_items)
df['AirlineNameCount'] = ccs.apply(lambda x: len(x))
#kontrola:   df.loc[25287, 'AirlineNameSummary']
```

Rámcový profil atributu AirlineNameSummary



CabinCodeSummary	Počet
coach	248532
first	449
coach|first	336
coach|premium coach	239
premium coach	219
business	154
business|coach	70
business|first	1
Celkem	**250000**

AirlineNameCount	Počet
1	224304
2	25403
3	293
Celkem	**250000**

- Atribut **EquipmentDescriptionSummary** jako shrnující atribut obsahující množinu typů letadel, které byly během daného letu využity
- Typy letadel rozděleny pro zjednodušení do 3 kategorií:
  - Airbus
  - Boeing
  - Other

New columns EquipmentDescriptionSummary

```
for i in range(250000):
    for j in range(df.columns.get_loc('1st_segmentEquipmentDescription'),
                   df.columns.get_loc('4th_segmentEquipmentDescription')+1):
        if pd.isnull(df.iloc[i,j]) == False:
            if 'airbus'.upper() in df.iloc[i,j].upper():
                df.iloc[i,j] = 'Airbus'
            elif 'boeing'.upper() in df.iloc[i,j].upper():
                df.iloc[i,j] = 'Boeing'
            else:
                df.iloc[i,j] = 'Other'
```

```
ccs = df.loc[:,['1st_segmentEquipmentDescription',
                '2nd_segmentEquipmentDescription',
                '3rd_segmentEquipmentDescription',
                '4th_segmentEquipmentDescription'
               ]].apply(remove_empty, axis=1)

df['EquipmentDescriptionSummary'] = ccs.apply(join_items)
```

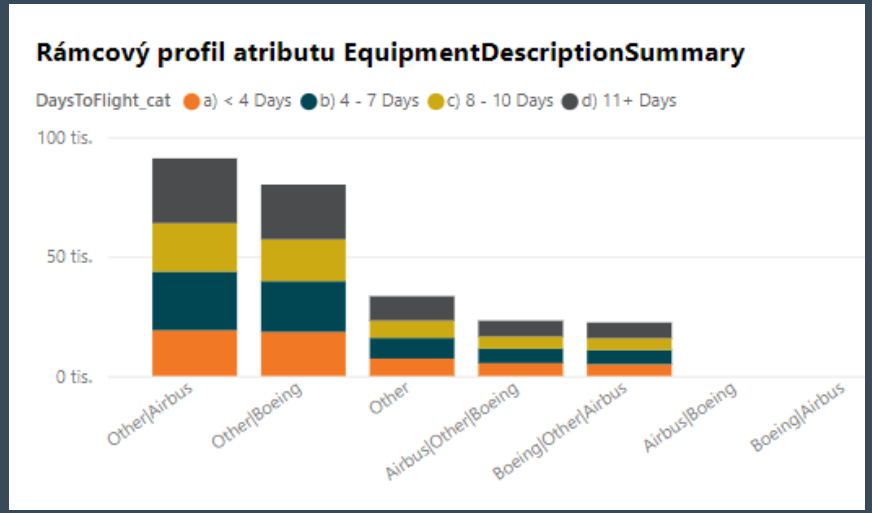
- Atribut **NumberOfTransfers** jako počet přestupů (odvozen jako počet neprázdných „_segment“ atributů minus 1)
- Atribut **DaysToFlight** jako počet dnů od zakoupení letenky do odletu (odvozen jako rozdíl atributů flightDate a searchDate)

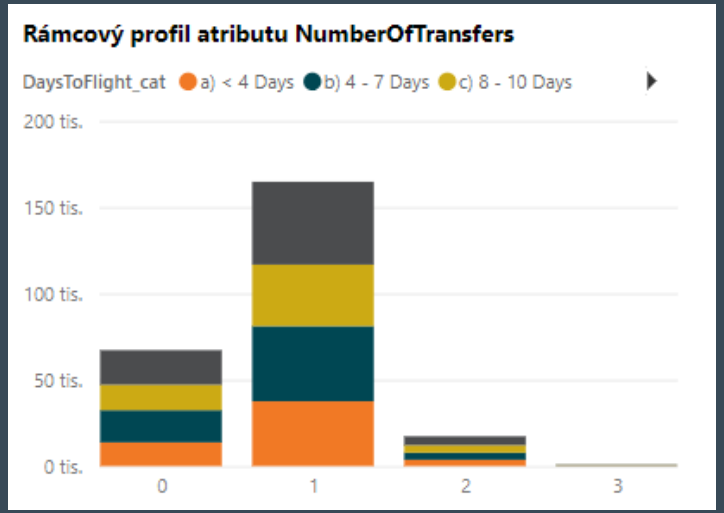
New column NumberOfTransfers

```
df['NumberOfTransfers']=df[['1st_segmentDepartureTimeEpochSeconds',
       '2nd_segmentDepartureTimeEpochSeconds',
       '3rd_segmentDepartureTimeEpochSeconds',
       '4th_segmentDepartureTimeEpochSeconds'
                          ]].apply(lambda row:
                                   sum(1 for cell in row if cell != '')-1,
                                   axis=1)
```

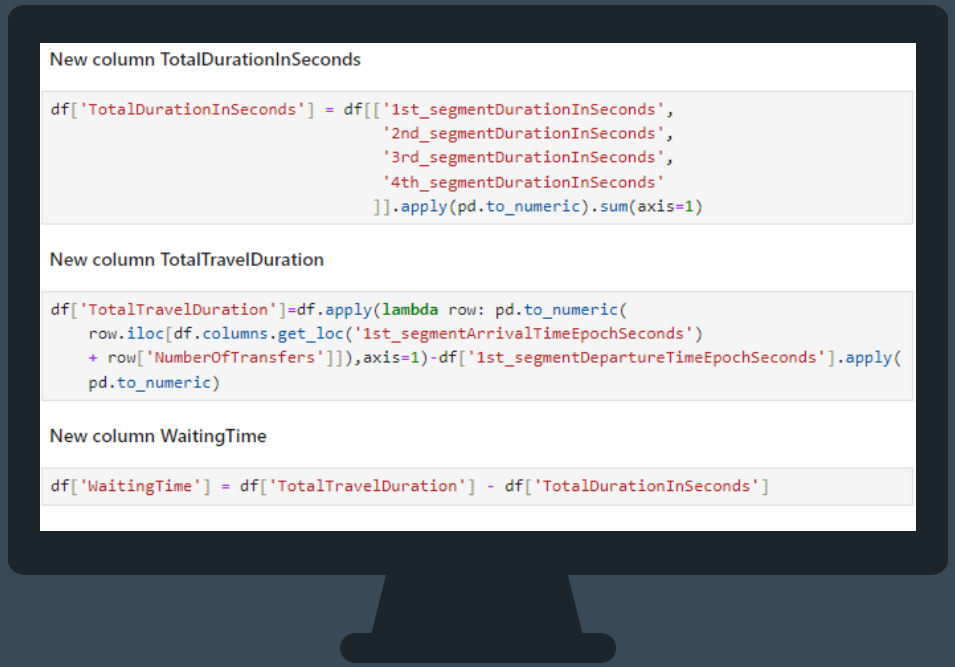
New column DaysToFlight

```
df['DaysToFlight'] = (df['flightDate'].apply(
    lambda x: datetime.strptime(x, '%Y-%m-%d'))
                      - df['searchDate'].apply(
                          lambda x: datetime.strptime(x, '%Y-%m-%d')
                      )).apply(lambda x: x.days)
```

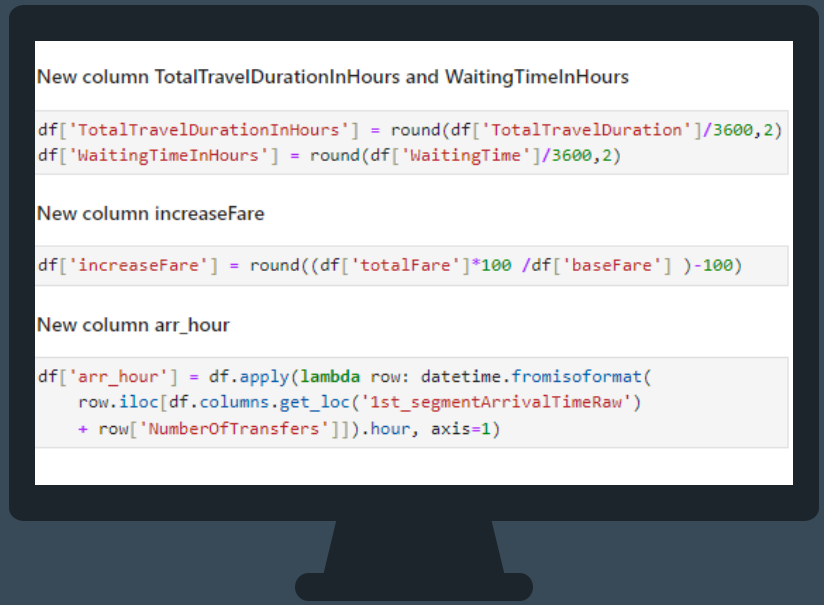

Rámcový profil atributu EquipmentDescriptionSummary
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
100 tis.
50 tis.
0 tis.
Other|Airbus
Other|Boeing
Other
Airbus|Other|Boeing
Boeing|Other|Airbus
Airbus|Boeing
Boeing|Airbus


Rámcový profil atributu NumberOfTransfers
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
200 tis.
150 tis.
100 tis.
50 tis.
0 tis.
0
1
2
3

- Atribut **TotalDurationInSeconds** jako celkový očekávaný čas strávený v letadle (odvozen jako součet očekávané doby letu všech dílčích letů)
- Atribut **TotalTravelDuration** jako celková očekávaná doba letu, včetně očekávaného čekání na přestupových letištích (odvozen jako rozdíl času očekávaného příletu do cílové destinace a času odletu z výchozí destinace)
- Atribut **WaitingTime** jako celková očekávaná doba čekání na přestupech (odvozen jako rozdíl předchozích dvou atributů)



- Atribut **TotalTravelDurationInHours** a **WaitingTimeInHours** jako celková očekávaná doba letu, resp. celková očekávaná doba čekání na přestupech, nyní převedeno na hodiny
- Atribut **increaseFare** jako relativní nárůst celkové ceny letenky (tj. včetně všech poplatků a daní) oproti její základní ceně (tj. bez poplatků a daní) v procentech
- Atribut **arr_hour** jako hodina příletu do cílové destinace



- Nové ID sloupce **a_dep_index** a **a_arr_index** umožňující následné napojení na tabulku Weather
  - „kód výchozího (resp. cílového) letiště & datum odletu (resp. příletu)“
- Vzhledem k tomu, že velká část atributů byla spojitého typu, bylo nutné pro potřeby pozdější analýzy (tj. pro Data Science část projektu) některé atributy kategorizovat.

New ID columns

```
df['arrivalDate'] = df.apply(
    lambda row: (datetime.strptime(row['flightDate'], '%Y-%m-%d')
                 + timedelta(days=row['elapsedDays'])
                ).strftime('%Y-%m-%d')
    if row['elapsedDays'] != 0 else row['flightDate'], axis=1)
```

```
df['a_dep_index'] = df['startingAirport'] + ' ' + df['flightDate']
df['a_arr_index'] = df['destinationAirport'] + ' ' + df['arrivalDate']
```

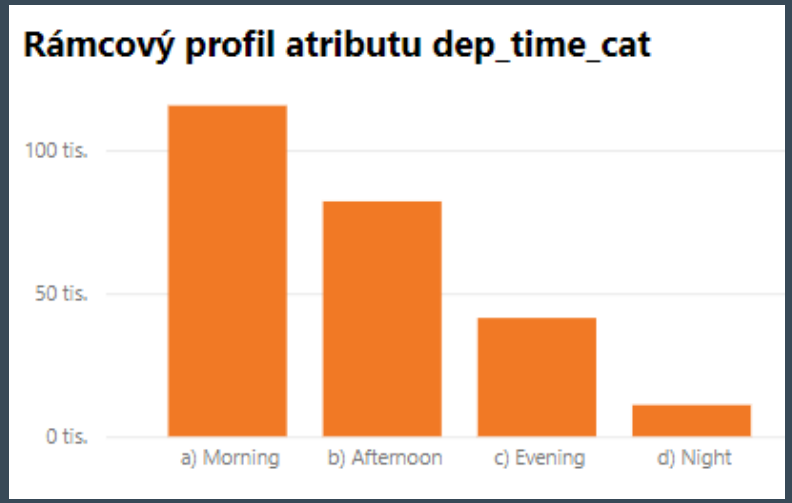
- Atribut **dep_time_cat** jako doba odletu
- Atribut **arr_time_cat** jako doba příletu
- Kategorie:
  - 00:00 – 05:59 → d) Night
  - 06:00 – 11:59 → a) Morning
  - 12:00 – 17:59 → b) Afternoon
  - 18:00 – 23:59 → c) Evening

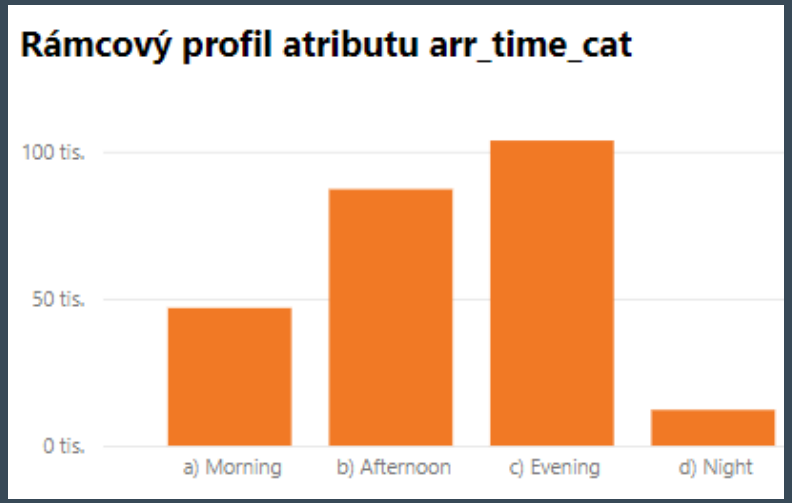
New column dep_time_cat

```
def extract_characters(text):
    if len(text) == 29:
        return int(text[11:13])
    else:
        return "error"

# Aplikace funkce na sloupec 1st_segmentDepartureTimeRaw
df['dep_time_cat']=pd.cut(df['1st_segmentDepartureTimeRaw'].apply(extract_characters),
                          bins = [-float("inf"),5,11,17,float("inf")],
                          labels = ['d) Night', 'a) Morning',
                                    'b) Afternoon', 'c) Evening'])
#hodina odletu na 1. prestupu
```

New column arr_time_cat

```
df['arr_time_cat'] = pd.cut(df['arr_hour'],
                            bins = [-float("inf"),5,11,17,float("inf")],
                            labels = ['d) Night', 'a) Morning',
                                      'b) Afternoon', 'c) Evening'])
```


Rámcový profil atributu dep_time_cat
100 tis.
50 tis.
0 tis.
a) Morning
b) Afternoon
c) Evening
d) Night


Rámcový profil atributu arr_time_cat
100 tis.
50 tis.
0 tis.
a) Morning
b) Afternoon
c) Evening
d) Night

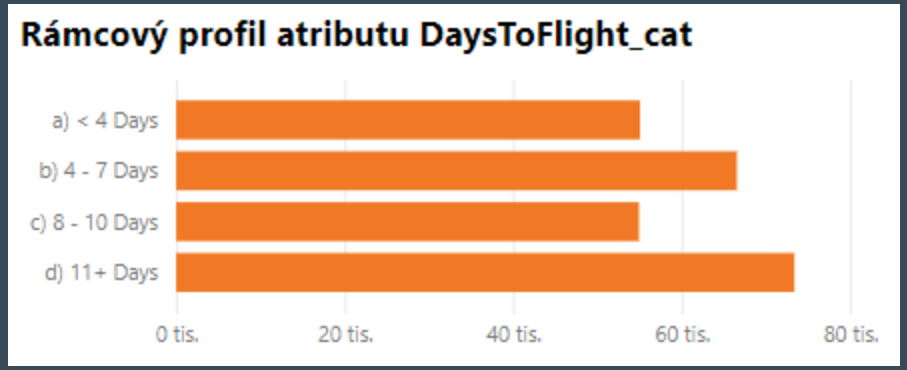
- Atribut **DaysToFlight_cat**
  - a) < 4 Days, b) 4 - 7 Days,
    c) 8 - 10 Days, d) 11+ Days
- Atribut **totalFare_cat** jako doba přílet
  - a) <200 USD, b) 200-300 USD,
    c) 300-400 USD, d) 400+ USD
- Atribut **increaseFare_cat** jako doba přílet
  - a) <13 %, b) 13-16 %,
    c) 16-21 %, d) 21+ %

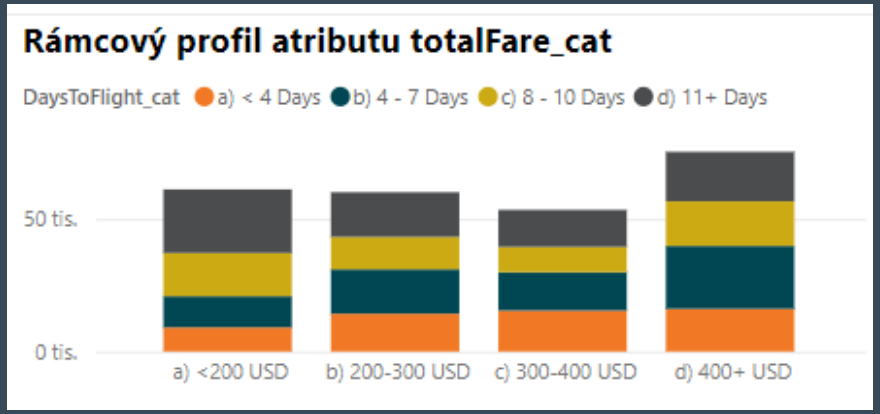
New column DaysToFlight_cat

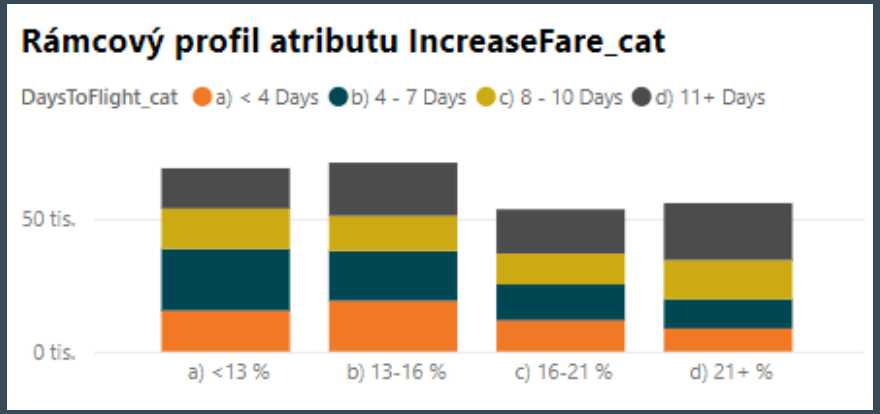
```
df['DaysToFlight_cat'] = pd.cut(df['DaysToFlight'],
                                bins = [-float("inf"),3,7,10, float("inf")],
                                labels = ['a) < 4 Days', 'b) 4 - 7 Days',
                                          'c) 8 - 10 Days', 'd) 11+ Days'])
```

New columns totalFare_cat a increaseFare_cat

```
df['totalFare_cat'] = pd.cut(df['totalFare'],
                             bins = [-float("inf"),200,300, 400,float("inf")],
                             labels = ['a) <200 USD', 'b) 200-300 USD',
                                       'c) 300-400 USD', 'd) 400+ USD'])
df['increaseFare_cat'] = pd.qcut(round((df['totalFare']*100 /df['baseFare'] )-100),
                                 4,
                                 labels = ['a) <13 %', 'b) 13-16 %',
                                           'c) 16-21 %', 'd) 21+ %'])
```


Rámcový profil atributu DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
0 tis.
20 tis.
40 tis.
60 tis.
80 tis.


Rámcový profil atributu totalFare_cat
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
50 tis.
0 tis.
a) <200 USD
b) 200-300 USD
c) 300-400 USD
d) 400+ USD


Rámcový profil atributu IncreaseFare_cat
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
50 tis.
0 tis.
a) <13 %
b) 13-16 %
c) 16-21 %
d) 21+ %

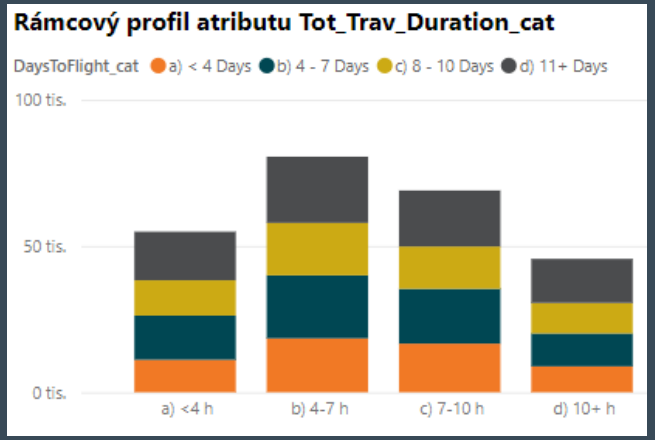
- Atribut **Tot_Trav_Duration_cat**
  - a) < 4 Hours, b) 4 - 7 Hours,
    c) 7 - 10 Hours, d) 10+ Hours
- Atribut **Waiting_Time_cat** jako doba přílet
  - a) 0 Hours, b) 0 - 2 Hours,
    c) 2+ Hours
- Atribut **totalTravelDistance_cat** jako doba přílet
  - a) <1000 Miles, b) 1000 – 2000 Miles,
    c) 2000+ Miles

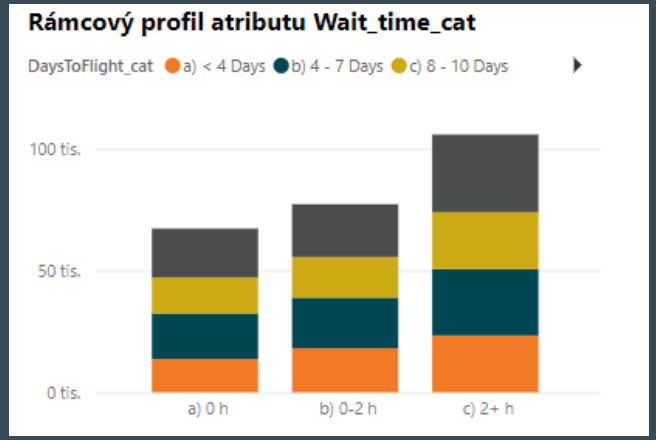
New columns Tot_Trav_Duration_cat and Wait_time_cat

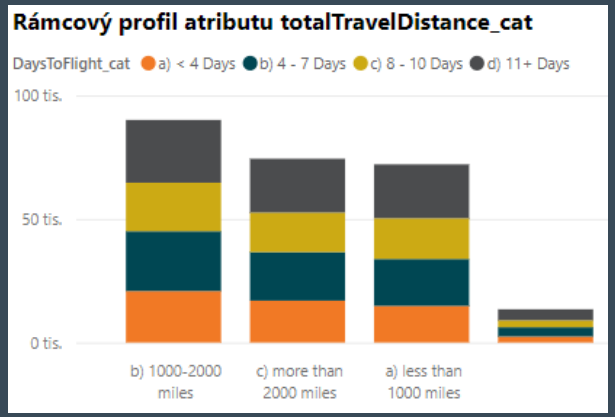
```
df['Tot_Trav_Duration_cat'] = pd.cut(df['TotalTravelDuration'],
                                     bins = [-float("inf"),14400,25200,36000, float("inf")],
                                     labels = ['a) <4 h', 'b) 4-7 h', 'c) 7-10 h', 'd) 10+ h'])
df['Wait_time_cat'] = pd.cut(df['WaitingTime'],
                             bins = [-float("inf"),0,7200, float("inf")],
                             labels = ['a) 0 h', 'b) 0-2 h', 'c) 2+ h'])
```

New column totalTravelDistance_cat

```
df['totalTravelDistance_cat'] = pd.cut(df['totalTravelDistance'],
                                       bins = [-float("inf"),1000,2000, float("inf")],
                                       labels = ['a) less than 1000 miles',
                                                 'b) 1000-2000 miles',
                                                 'c) more than 2000 miles'])
```


Rámcový profil atributu Tot_Trav_Duration_cat
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
100 tis.
50 tis.
0 tis.
a) <4 h
b) 4-7 h
c) 7-10 h
d) 10+ h


Rámcový profil atributu Wait_time_cat
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
100 tis.
50 tis.
0 tis.
a) 0 h
b) 0-2 h
c) 2+ h


Rámcový profil atributu totalTravelDistance_cat
DaysToFlight_cat
a) < 4 Days
b) 4 - 7 Days
c) 8 - 10 Days
d) 11+ Days
100 tis.
50 tis.
0 tis.
b) 1000-2000 miles
c) more than 2000 miles
a) less than 1000 miles

- Na závěr byly z výsledné tabulky promazány atributy, které nebudou dále využity.

```
df.drop(labels=['legId', 'fareBasisCode', 'segmentsAirlineCode', 'segmentsDepartureTimeEpochSeconds',
                'segmentsDepartureTimeRaw', 'segmentsArrivalTimeEpochSeconds', 'segmentsArrivalTimeRaw',
                'segmentsArrivalAirportCode', 'segmentsDepartureAirportCode', 'segmentsAirlineName',
                'segmentsEquipmentDescription', 'segmentsDurationInSeconds', 'segmentsDistance',
                'segmentsCabinCode'], axis = 'columns', inplace = True)
```

```
df.drop(labels=['isRefundable',
'1st_segmentDepartureTimeEpochSeconds','2nd_segmentDepartureTimeEpochSeconds',
                '3rd_segmentDepartureTimeEpochSeconds','4th_segmentDepartureTimeEpochSeconds',
'1st_segmentDepartureTimeRaw','2nd_segmentDepartureTimeRaw',
                '3rd_segmentDepartureTimeRaw','4th_segmentDepartureTimeRaw',
'1st_segmentArrivalTimeEpochSeconds','2nd_segmentArrivalTimeEpochSeconds',
                '3rd_segmentArrivalTimeEpochSeconds','4th_segmentArrivalTimeEpochSeconds',
'1st_segmentArrivalTimeRaw','2nd_segmentArrivalTimeRaw',
                '3rd_segmentArrivalTimeRaw','4th_segmentArrivalTimeRaw',
'1st_segmentArrivalAirportCode','2nd_segmentArrivalAirportCode',
                '3rd_segmentArrivalAirportCode','4th_segmentArrivalAirportCode',
'1st_segmentDepartureAirportCode','2nd_segmentDepartureAirportCode',
                '3rd_segmentDepartureAirportCode','4th_segmentDepartureAirportCode',
'1st_segmentAirlineName','2nd_segmentAirlineName',
                '3rd_segmentAirlineName','4th_segmentAirlineName',
'1st_segmentEquipmentDescription','2nd_segmentEquipmentDescription',
                '3rd_segmentEquipmentDescription','4th_segmentEquipmentDescription',
'1st_segmentDistance','2nd_segmentDistance','3rd_segmentDistance','4th_segmentDistance',
'1st_segmentCabinCode','2nd_segmentCabinCode','3rd_segmentCabinCode','4th_segmentCabinCode',
'travelDuration', 'arr_hour'], axis = 'columns', inplace = True)
```

- Ponechané atributy byly uloženy do nového csv souboru.

```
df.columns
```

```
Index(['searchDate', 'flightDate', 'startingAirport', 'destinationAirport',
       'elapsedDays', 'isBasicEconomy', 'isNonStop', 'baseFare', 'totalFare',
       'seatsRemaining', 'totalTravelDistance', '1st_segmentDurationInSeconds',
       '2nd_segmentDurationInSeconds', '3rd_segmentDurationInSeconds',
       '4th_segmentDurationInSeconds', 'CabinCodeSummary',
       'AirlineNameSummary', 'AirlineNameCount', 'EquipmentDescriptionSummary',
       'NumberOfTransfers', 'DaysToFlight', 'TotalDurationInSeconds',
       'TotalTravelDuration', 'WaitingTime', 'increaseFare', 'dep_time_cat',
       'arr_time_cat', 'DaysToFlight_cat', 'Tot_Trav_Duration_cat',
       'Wait_time_cat', 'totalTravelDistance_cat', 'totalFare_cat',
       'increaseFare_cat', 'arrivalDate', 'a_dep_index', 'a_arr_index',
       'TotalTravelDurationInHours', 'WaitingTimeInHours'],
      dtype='object')
```

```
df.to_csv(path_or_buf='itineraries_250k_final3.csv')
```

ANALÝZA DAT

Analýza v Power BI

CENY LETENEK I

- Průměrné ceny letenek bývají obecně nejnižší při velkém/velmi malém množství zbývajících volných míst.
- Obzvlášť vysoké byly ceny letenek v neděli 24.4.2022.
    - Vysoké ceny letenek mohly souviset např. s jarními prázdninami.

CENY LETENEK II

- Nová míra totalXbase
  - Kolikrát vzrostla hodnota atributu totalFare oproti příslušné hodnotě baseFare.
  - V průměru byla celková cena letenky oproti její původní ceně cca 1,16x vyšší, průměr nejvíce zvyšovaly letenky do třídy coach

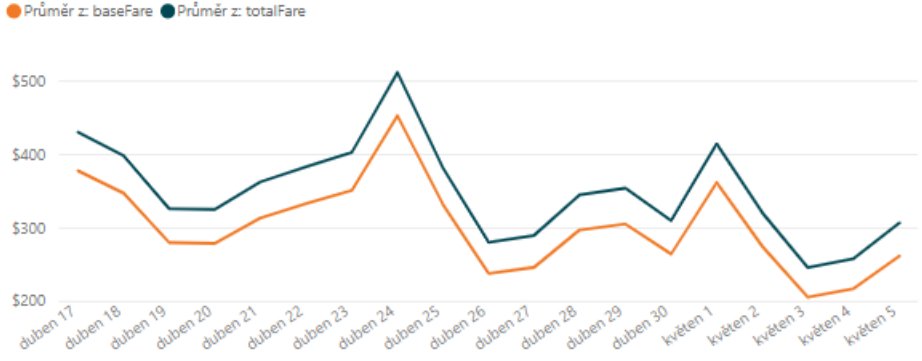
CENY LETENEK III

- Obecně bývají nejlevnější letenky uprostřed týdne (úterý, středa), nejdražší naopak kolem víkendu (především neděle, ale například také pondělí atd.).

CabinCodeSummary

business	business|coach	business|first	coach	coach|first	coach|premium coach	first	premium coach

Průměr z: baseFare a Průměr z: totalFare podle kategorie Měsíc a Den

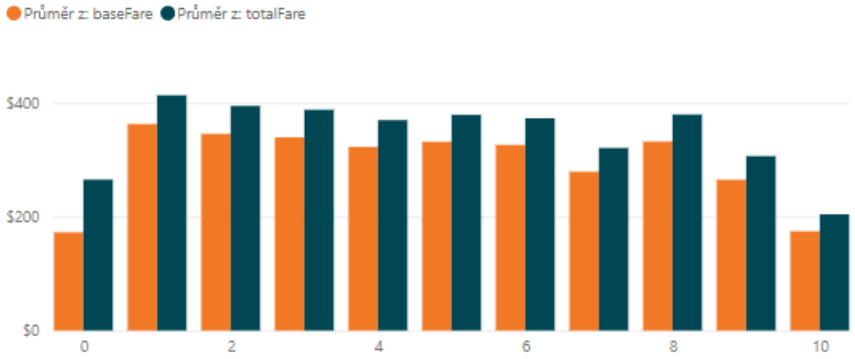


$302
Průměr z: baseFare

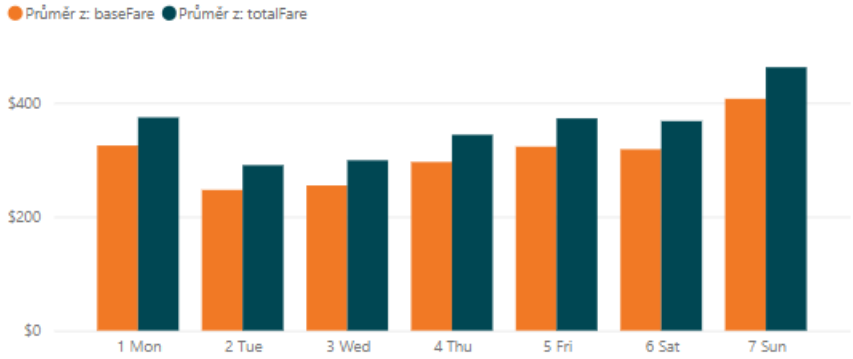
$350
Průměr z: totalFare

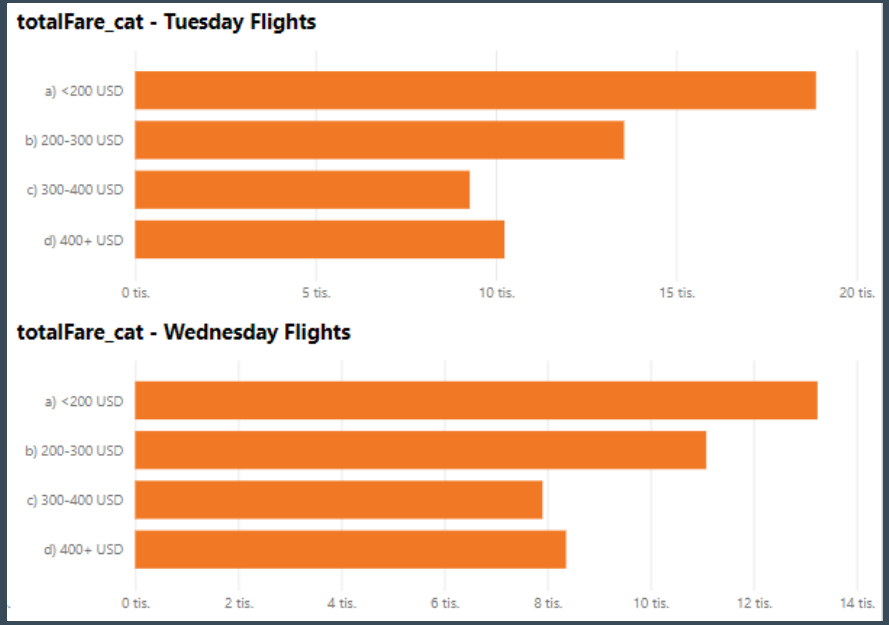
1,16
totalXbase

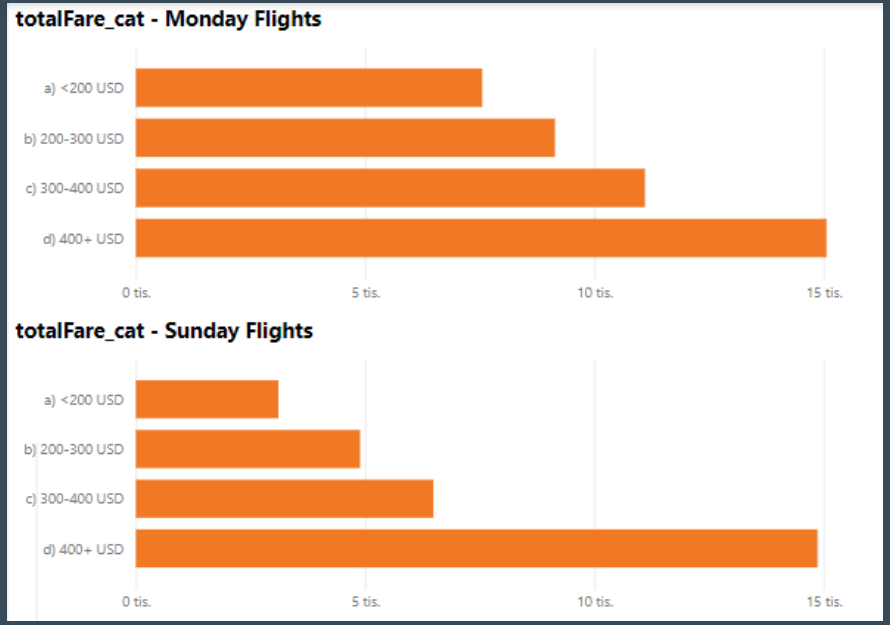
Průměr z: baseFare a Průměr z: totalFare podle kategorie seatsRemaining



Průměr z: baseFare a Průměr z: totalFare podle kategorie Weekday




totalFare_cat - Tuesday Flights
a) <200 USD
b) 200-300 USD
c) 300-400 USD
d) 400+ USD
0 tis.
5 tis.
10 tis.
15 tis.
20 tis.
totalFare_cat - Wednesday Flights
a) <200 USD
b) 200-300 USD
c) 300-400 USD
d) 400+ USD
0 tis.
2 tis.
4 tis.
6 tis.
8 tis.
10 tis.
12 tis.
14 tis.


totalFare_cat - Monday Flights
a) <200 USD
b) 200-300 USD
c) 300-400 USD
d) 400+ USD
0 tis.
5 tis.
10 tis.
15 tis.
totalFare_cat - Sunday Flights
a) <200 USD
b) 200-300 USD
c) 300-400 USD
d) 400+ USD
0 tis.
5 tis.
10 tis.
15 tis.

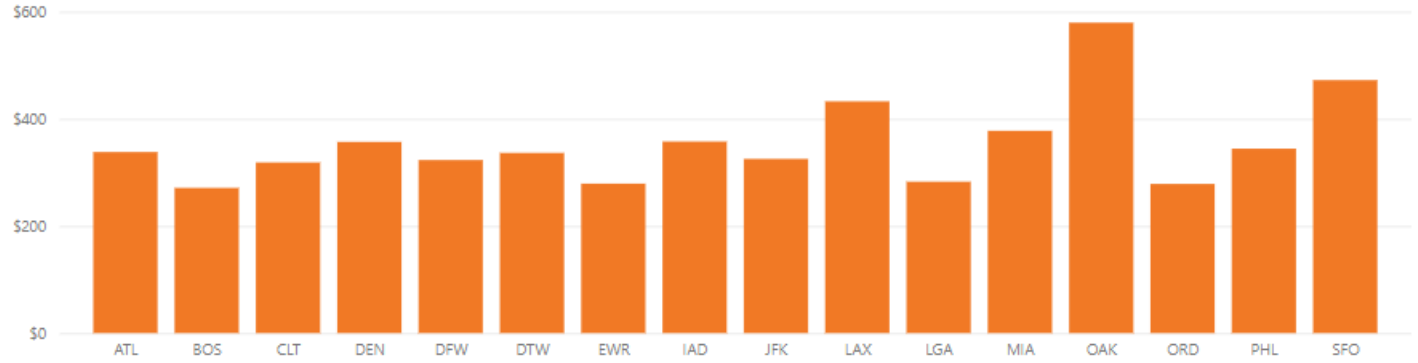
CENY LETENEK IV

- Nejdražší bývají letenky z/do Oaklandu, a to nezávisle na dni v týdnu.
- Dále bývají poměrně drahé i letenky z Los Angeles nebo San Francisca, v soboty a neděle navíc také letenky z Miami.

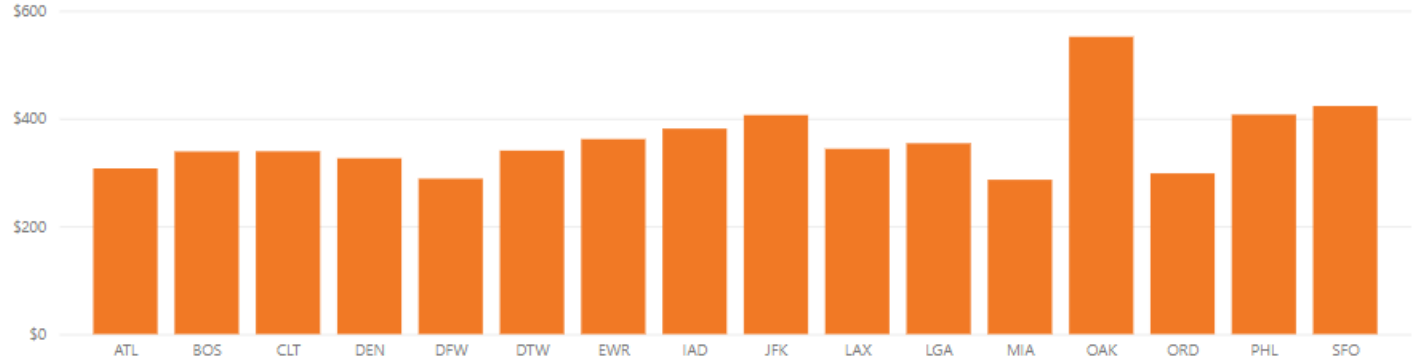
Weekday

1 Mon	2 Tue	3 Wed	4 Thu	5 Fri	6 Sat	7 Sun

Průměr z: totalFare podle kategorie startingAirport



Průměr z: totalFare podle kategorie destinationAirport



Airport_Code	Airport_City
ATL	Atlanta
BOS	Boston
CLT	Charlotte
DEN	Denver
DFW	Dallas
DTW	Detroit
EWR	Newark
IAD	Dulles, DC
JFK	New York
LAX	Los Angeles
LGA	New York
MIA	Miami
OAK	Oakland
ORD	Chicago
PHL	Philadelphia
SFO	San Francisco

$350

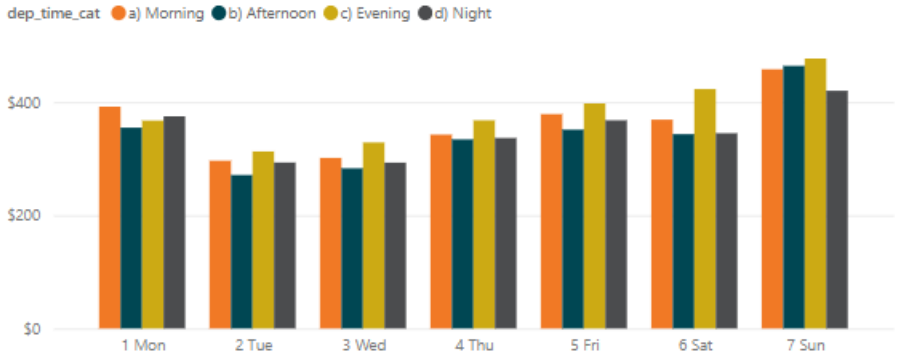
Průměr z: totalFare

CENY LETENEK V

- Večerní lety bývají drahé, odpolední lety naopak patří mezi ty levnější.
- Ceny letenek se vyšplhají na maximum obvykle 4-7 dní před odletem.
- Mezi průměrnou cenou letenky a počtem přestupů platí přímá úměrnost.

TotalFare & WeekDay, SeatsRemaining, NumberOfTransfers

Průměr z: totalFare podle kategorie Weekday a dep_time_cat



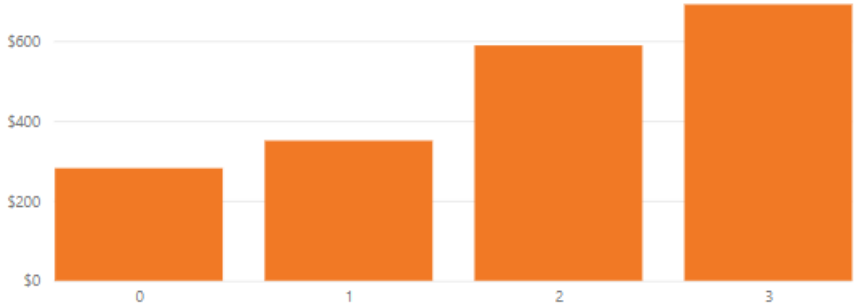
$350

Průměr z: totalFare

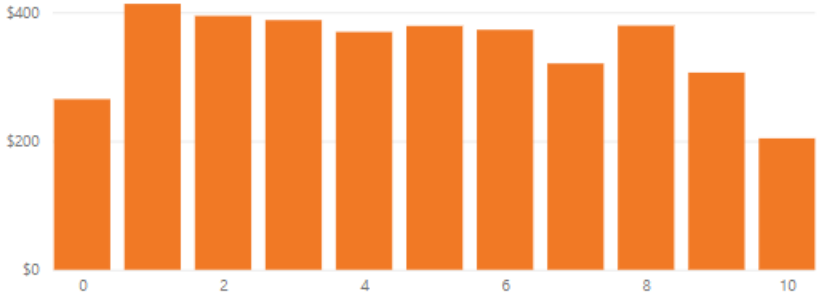
250 tis.

Počet letenek

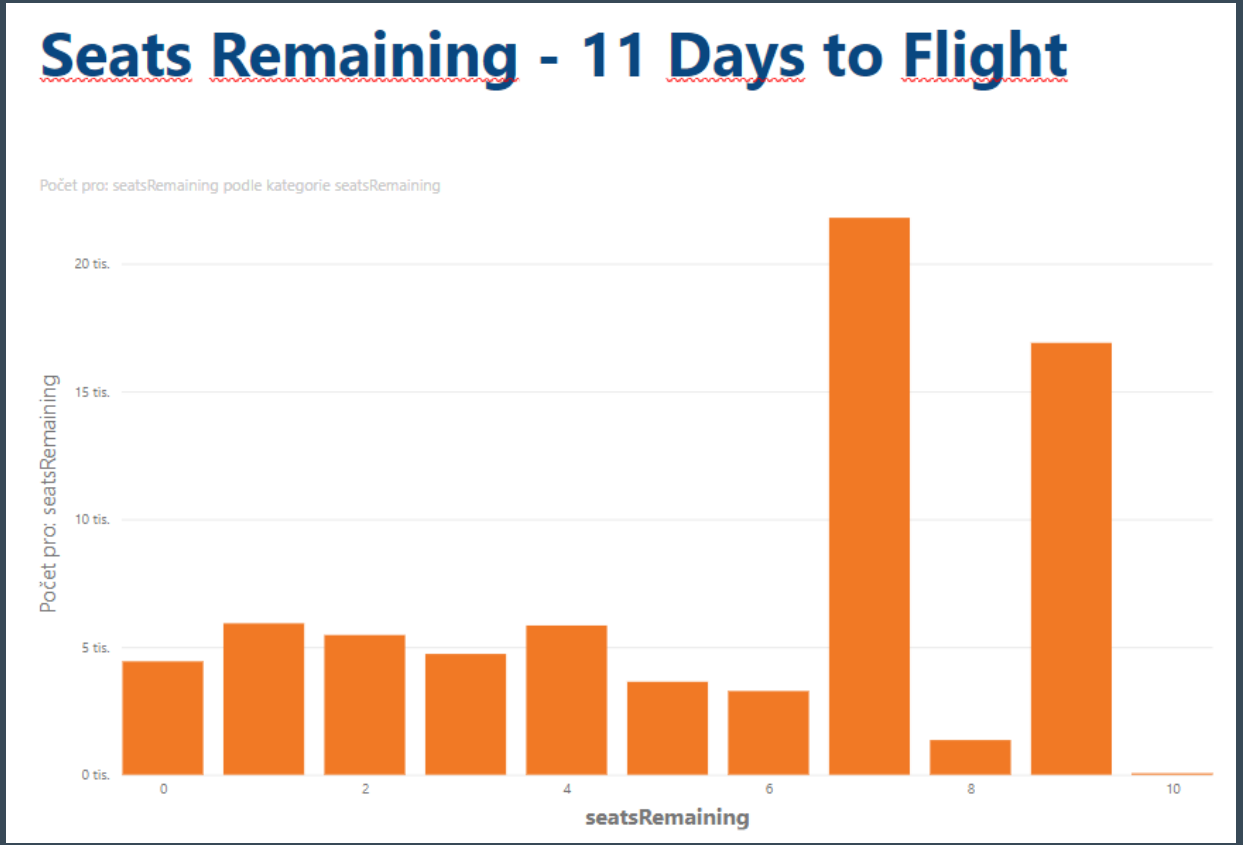
Průměr z: totalFare podle kategorie NumberOfTransfers



Průměr z: totalFare podle kategorie seatsRemaining

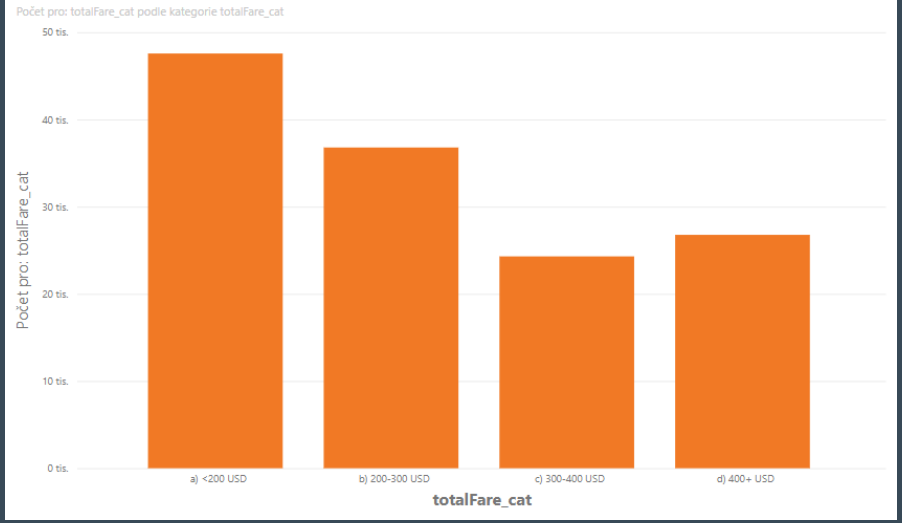


Zbývá-li dostatečný počet dnů do odletu, bývá obvykle počet volných míst v letadle ještě dostatek.

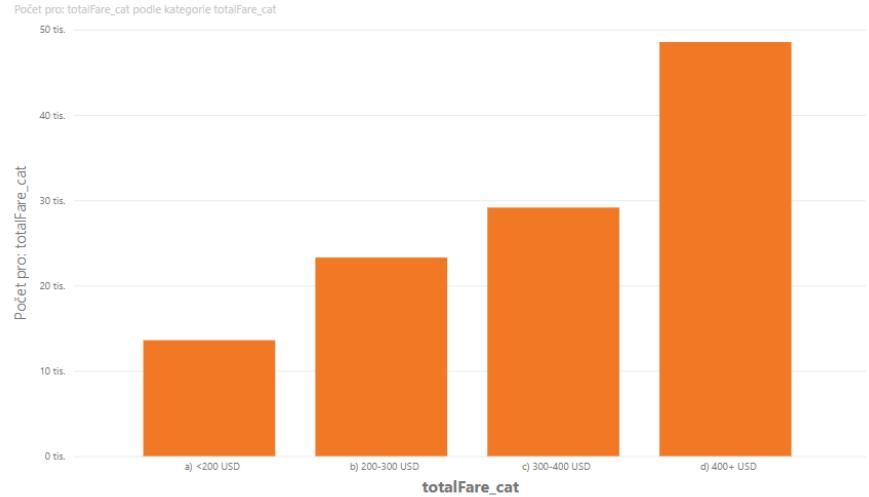


Kratší lety obvykle spadají do nižších cenových kategorií, cesty s delší dobou trvání bývají naopak dražší.

Total Fare Category - less than 7 hours flight



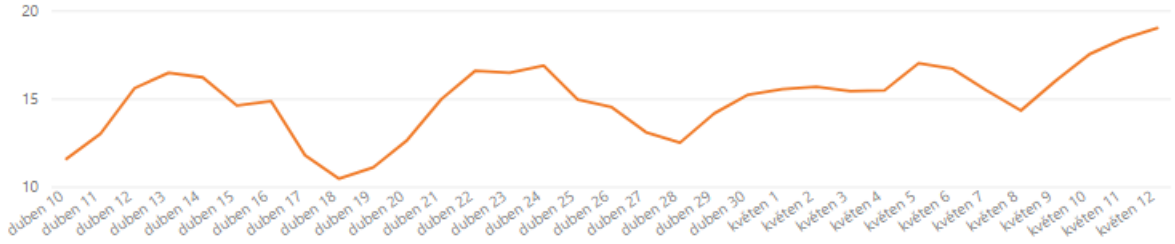
Total Fare Category - more than 7 hours flight



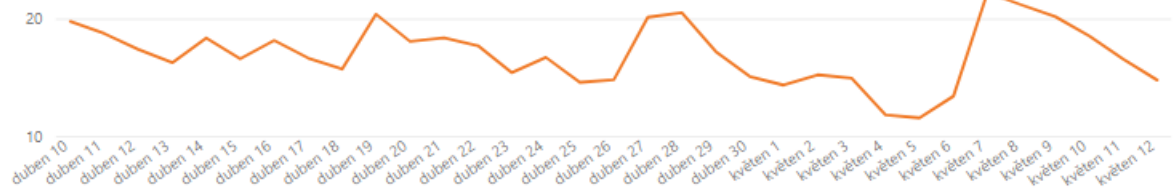
POČASÍ

- Nejvyšší teploty na jihovýchodě USA (Dallas, Atlanta, Charlotte a především Miami)
- Největší úhrn srážen byl v Miami a Chicagu, nejmenší naopak na váchodě USA.

Průměr z: tavg podle kategorie Měsíc a Den



Průměr z: wspd podle kategorie Měsíc a Den



Průměr z: prcp podle kategorie Měsíc a Den

Airport_City

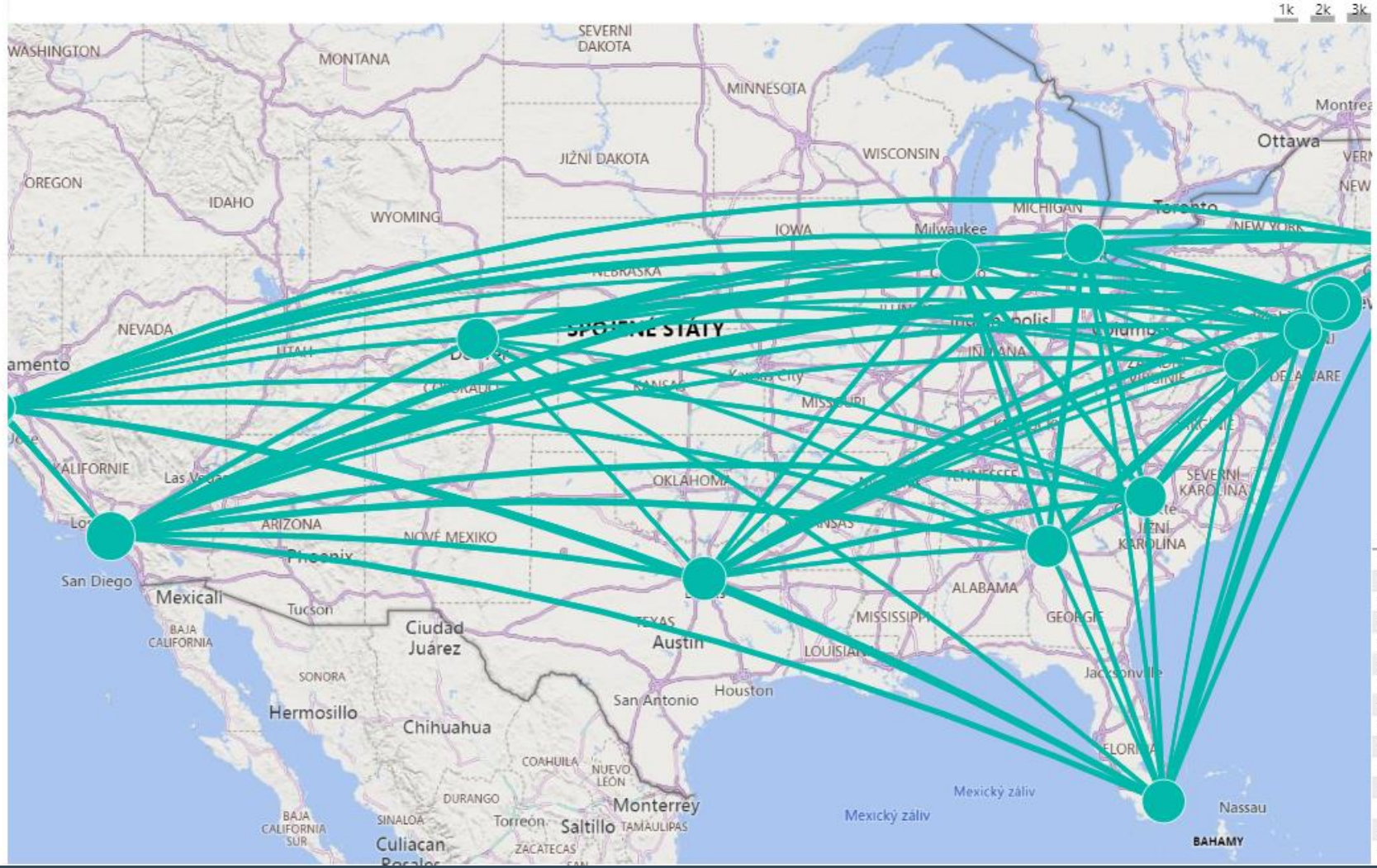
Atlanta	Dulles, DC	New York
Boston	Charlotte	Newark
Dallas	Chicago	Oakland
Denver	Los Angeles	Philadelphia
Detroit	Miami	San Francisco

14,96
Průměr z: tavg

2,27
Průměr z: prcp

FLOW MAP

- Pro lepší představu o geografické poloze jednotlivých letišť byla vykreslena rovněž mapa zobrazující jednotlivá města a trasy mezi nimi.


1k
2k
3k
WASHINGTON
MONTANA
SEVERNÍ DAKOTA
MINNESOTA
JIŽNÍ DAKOTA
WISCONSIN
Ottawa
OREGON
IDAHO
WYOMING
MICHIGAN
Toronto
IOWA
Milwaukee
NEBRASKA
NEVADA
SPOJENÉ STÁTY
INDIANA
KALIFORNIE
OKLAHOMA
SEVERNÍ KAROLÍNA
JIŽNÍ KAROLÍNA
ARIZONA
NOVÉ MEXIKO
Phoenix
San Diego
Mexicali
Tucson
ALABAMA
MISSISSIPPI
GEORGIE
Ciudad Juárez
TEXAS
Austin
LOUISIANA
Houston
San Antonio
BAJA CALIFORNIA
SONORA
Hermosillo
Chihuahua
COAHUILA
NUEVO LEÓN
DURANGO
Monterrey
Torreón
Saltillo
TAMAULIPAS
SINALOA
BAJA CALIFORNIA SUR
Culiacan
ZACATECAS
Mexický záliv
Nassau
BAHAMY
Original_Airport_City
Atlanta
Boston
Dallas
Denver
Detroit
Dulles, DC
Charlotte
Chicago
Los Angeles
Miami
New York
Newark
Oakland
Philadelphia
San Francisco
6,93
Hours (Avg)
Destination_Airport_City
Atlanta
Boston
Dallas
Denver
Detroit
Dulles, DC
Charlotte
Chicago
Los Angeles
Miami
New York
Newark
Oakland
Philadelphia
San Francisco

ZÁVĚR BI ANALÝZY I

- V rámci BI analýzy se podařilo odhalit atributy významně ovlivňující cenu letenky.
  - Den v týdnu
  - Počáteční/cílová destinace
  - Délka cesty
  - Počet dní do odletu
  - Denní doba odletu
  - Počet zbývajících míst

ZÁVĚRY BI ANALÝZY II

- V další části projektu bude cílem mimo jiné blíže specifikovat vliv těchto atributů na celkovou cenu letenky či hledat takové podmínky, při kterých se ceny letenek výrazně liší od již nalezených trendů.

DATA SCIENCE

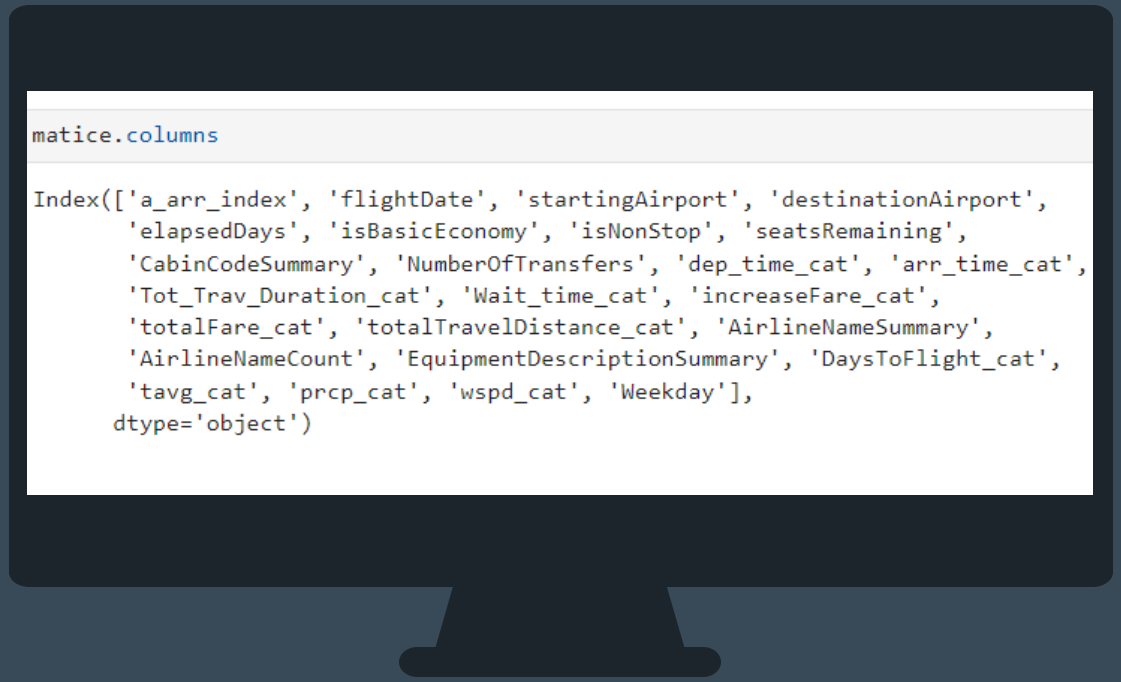
Detailnější zkoumání souvislostí mezi atributy na základě výsledků předchozí analýzy v Power BI

Příprava dat pro Cleverminer

- Potřebné atributy byly sloučeny do jedné datové matice

```
weather = pd.read_csv('weather_airports_hlavni.csv')
weather.drop(labels=['Unnamed: 0','time', 'tavg', 'tmin', 'tmax', 'prcp', 'snow',
       'wdir', 'wspd', 'wpgt', 'pres', 'tsun', 'Airport_Code'],
            axis = 'columns', inplace = True)
weekdays = pd.read_csv('weekdays.csv')
df = pd.read_csv('itineraries_250k_final3.csv', low_memory= False )
df = df[['a_dep_index', 'a_arr_index', 'flightDate',
       'startingAirport', 'destinationAirport', 'elapsedDays',
       'isBasicEconomy', 'isNonStop',
       'seatsRemaining',
       'CabinCodeSummary', 'NumberOfTransfers',
       'dep_time_cat', 'arr_time_cat', 'Tot_Trav_Duration_cat',
       'Wait_time_cat', 'increaseFare_cat',
       'totalFare_cat', 'totalTravelDistance_cat', 'AirlineNameSummary',
       'AirlineNameCount', 'EquipmentDescriptionSummary', 'DaysToFlight_cat']]
matice = pd.merge(df, weather, how="left", left_on='a_arr_index', right_on ='w_index')
matice = pd.merge(matice, weekdays, how="left", left_on='flightDate', right_on ='Date')
matice.drop(labels=['a_dep_index', 'w_index', 'Date'], axis = 'columns', inplace = True)
```

- Pro následnou analýzu v Clevermineru byly použity vybrané atributy.



```
matice.columns
```

```
Index(['a_arr_index', 'flightDate', 'startingAirport', 'destinationAirport',
       'elapsedDays', 'isBasicEconomy', 'isNonStop', 'seatsRemaining',
       'CabinCodeSummary', 'NumberOfTransfers', 'dep_time_cat', 'arr_time_cat',
       'Tot_Trav_Duration_cat', 'Wait_time_cat', 'increaseFare_cat',
       'totalFare_cat', 'totalTravelDistance_cat', 'AirlineNameSummary',
       'AirlineNameCount', 'EquipmentDescriptionSummary', 'DaysToFlight_cat',
       'tavg_cat', 'prcp_cat', 'wspd_cat', 'Weekday'],
      dtype='object')
```

SEZNAM ANALYTICKÝCH OTÁZEK I

- Na základě předchozí BI analýzy byly sestaveny analytické otázky, kterým se bude tato část práce věnovat.
- *Bylo by vhodné na některých trasách přidat lety?*
- *Existují lety, které jsou i přes svou krátkou dobu trvání drahé?*

SEZNAM ANALYTICKÝCH OTÁZEK II

- *Existují naopak lety, které jsou i přes svou větší délku nezvykle levné?*
- *Lze nalézt takové podmínky, při kterých je cena letenky výrazně nižší/vyšší, než bývá v daný den v týdnu obvyklé?*
- *Existují trasy, na kterých obecně levnější aerolinka létá nezvykle draze oproti jiným aerolinkám?*

SEZNAM ANALYTICKÝCH OTÁZEK III

- *A existují naopak trasy, na kterých některá letecká společnost výší cen svých letenek převyšuje i aerolinku s obvykle dražšími letenkami?*
- Kromě sepsaných analytických otázek bude v této části projektu rovněž sestaven model pro predikci celkové ceny letenky.

01

4FT-MINER

I) POSÍLENÍ NĚKTERÝCH VYTÍŽENÝCH SPOJŮ?

- *Bylo by vhodné na některých trasách přidat lety?*
- Cílem této analýzy bude nalezení takových tras, které bývají z větší části vyprodané už delší dobu před odletem.
- Jinými slovy, existuje nějaká trasa, na které se spolehlivostí alespoň 40 % budou 11 dní před odletem zbývat maximálně 2 poslední volná sedadla splňující zákazníkem zadané parametry?

- Vzhledem k povaze otázky byla pro sukcedent seatsRemaining zvolena varianta ‚lcut‘ a pro antecedent DaysToFlight_cat bude brána v potaz pouze kategorie ‚d) 11+ Days‘

```
clm = cleverminer(df=matice,proc='4ftMiner',
              quantifiers= {'conf':0.4, 'Base':50},
              ante ={
                   'attributes':[
                       {'name': 'DaysToFlight_cat', 'type':'one', 'value': 'd) 11+ Days'},
                       {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen':1},
                       {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                   ], 'minlen':3, 'maxlen':3, 'type':'con'},
              succ ={
                   'attributes':[
                       {'name': 'seatsRemaining', 'type':'lcut', 'minlen': 1, 'maxlen':2},
                   ], 'minlen':1, 'maxlen':1, 'type':'con'},
              )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
CleverMiner task processing summary:

Task type : 4ftMiner
Number of verifications : 507
Number of rules : 1
Total time needed : 00h 00m 02s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 00s


List of rules:
RULEID BASE  CONF  AAD    Rule
     1   104 0.446 +1.531 DaysToFlight_cat(d) 11+ Days) & startingAirport(DTW) & destinationAirport(EWR) => seatsRemaining(0 1) | ---



Rule id : 1

Base :   104  Relative base : 0.000  CONF : 0.446  AAD : +1.531  BAD : -1.531

Cedents:
  antecedent : DaysToFlight_cat(d) 11+ Days) & startingAirport(DTW) & destinationAirport(EWR)
  succcedent : seatsRemaining(0 1)
  condition  : ---

Fourfold table
    |  S  |  ¬S |
----|-----|-----|
 A  |  104|  129|
----|-----|-----|
¬A  |43982|205785|
----|-----|-----|
```

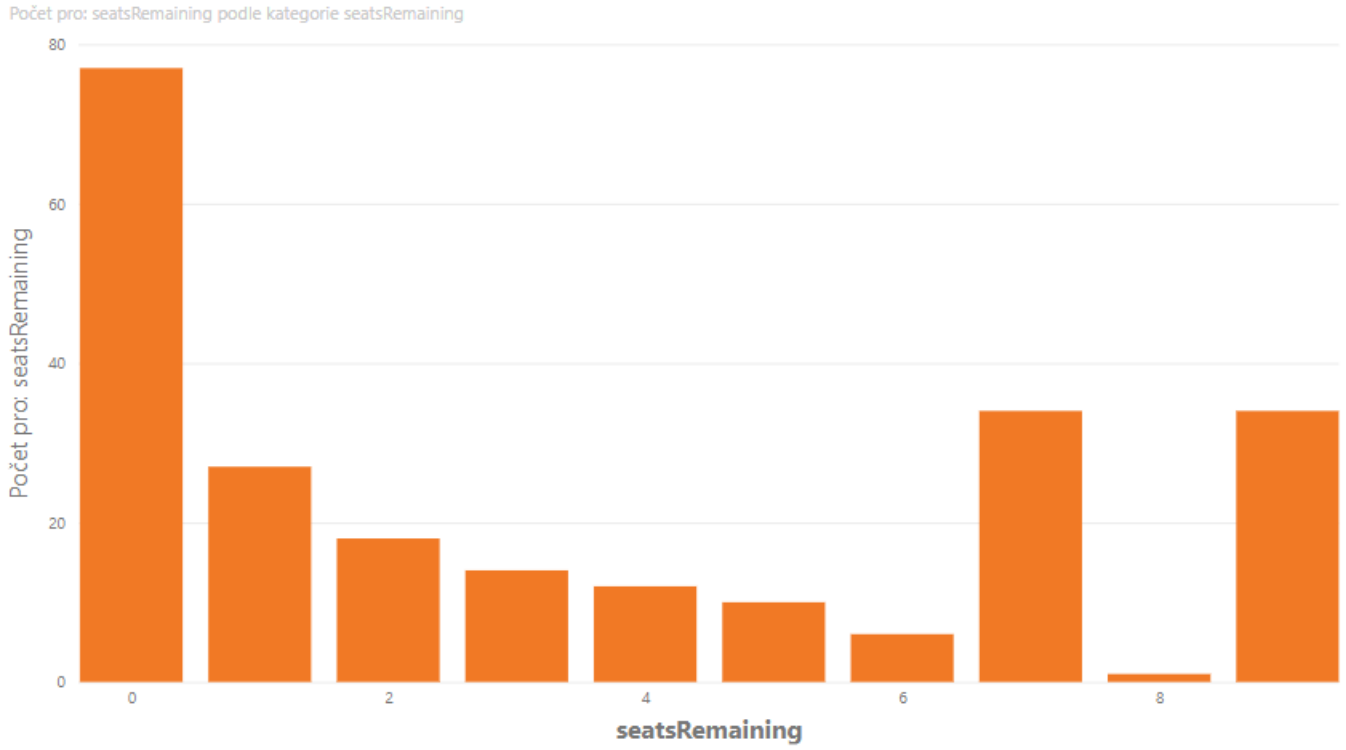
- Konfidence = 104/(104+129) = = 0,446
- 44,6 % letenek splňujících DaysToFlight_cat(d) 11+ Days) & startingAirport(DTW) & destinationAirport(EWR) splňuje i seatsRemaining(0 1)

POSÍLENÍ VYTÍŽENÝCH SPOJŮ – INTERPRETACE

- Získané pravidlo říká, že téměř 45 % spojů z Detroitu do Newarku bude mít už 11 dní předem téměř vyprodáno (0 nebo 1 zvývajících míst)

- Bylo by vhodné zvážit posílené této trasy, zjevně je po ní vysoká poptávka.

Seats Remaining - 11 Days to Flight



startingAirport

ATL	DFW	MIA	SFO
BOS	DTW	OAK	
CLT	IAD	ORD	
DEN	LAX	PHL	

destinationAirport

ATL	DFW	LAX	ORD
BOS	EWR	LGA	PHL
CLT	IAD	MIA	SFO
DEN	JFK	OAK	

IIa) KRÁTKÉ & DRAHÉ LETY

- *Existují lety, které trvají maximálně 7 hodin, ale i přesto patří alespoň 70 % z nich do nejdražší cenové kategorie, tzn. jejich cena včetně všech poplatků přesáhne hranici 400 USD?*

- Vzhledem k povaze otázky byla pro sukcedent totalFare_cat zvolena varianta ‚rcut‘ a pro antecedent Tot_Trav_Duration_cat bude brána varianta ‚lcut‘ maximální délky 2

```
# drahe kratsi lety
clm = cleverminer(df=matice,proc='4ftMiner',
                  quantifiers= {'conf':0.7, 'Base':50},
                  ante ={
                       'attributes':[
                           {'name': 'Tot_Trav_Duration_cat', 'type':'lcut', 'minlen': 1, 'maxlen':2},
                           {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen':1},
                           {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                       ], 'minlen':3, 'maxlen':5, 'type':'con'},
                  succ ={
                       'attributes':[
                           {'name': 'totalFare_cat', 'type':'rcut', 'minlen': 1, 'maxlen':1},
                       ], 'minlen':1, 'maxlen':1, 'type':'con'},
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

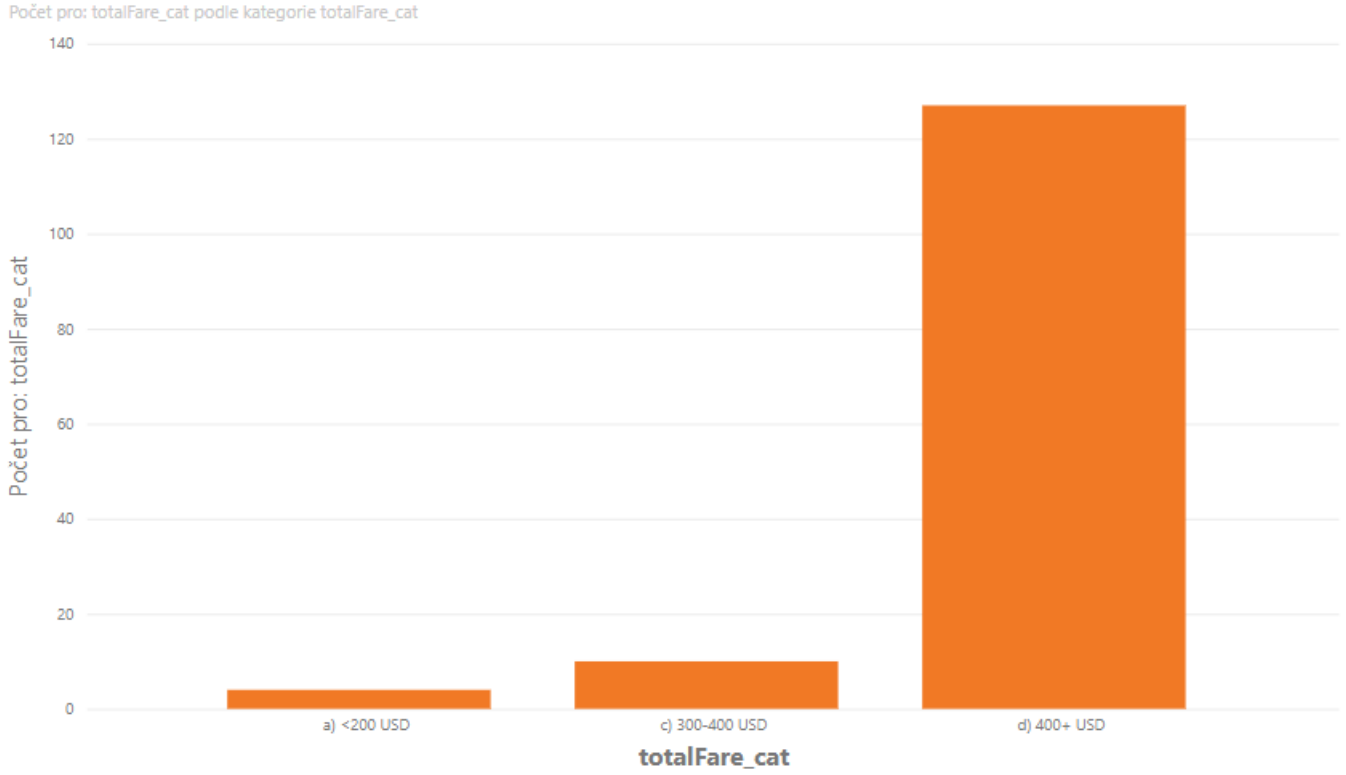
```
Task type : 4ftMiner
Number of verifications : 212
Number of rules : 11
Total time needed : 00h 00m 02s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 00s


List of rules:
RULEID BASE  CONF  AAD    Rule
     1   127 0.901 +1.990 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(ATL) & destinationAirport(SFO) => totalFare_cat(d) 400+ USD) | ---
     2    67 0.798 +1.648 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(CLT) & destinationAirport(SFO) => totalFare_cat(d) 400+ USD) | ---
     3   189 0.867 +1.878 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(DTW) & destinationAirport(SFO) => totalFare_cat(d) 400+ USD) | ---
     4   199 0.881 +1.923 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(IAD) & destinationAirport(SFO) => totalFare_cat(d) 400+ USD) | ---
     5    68 0.819 +1.720 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(PHL) & destinationAirport(SFO) => totalFare_cat(d) 400+ USD) | ---
     6   277 0.858 +1.847 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(ATL) => totalFare_cat(d) 400+ USD) | ---
     7    95 0.848 +1.816 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(CLT) => totalFare_cat(d) 400+ USD) | ---
     8   239 0.888 +1.949 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(DTW) => totalFare_cat(d) 400+ USD) | ---
     9   213 0.789 +1.619 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(IAD) => totalFare_cat(d) 400+ USD) | ---
    10   253 0.727 +1.413 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(OAK) => totalFare_cat(d) 400+ USD) | ---
    11    76 0.704 +1.336 Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(SFO) & destinationAirport(PHL) => totalFare_cat(d) 400+ USD) | ---


Rule id : 1

Base :   127  Relative base : 0.001  CONF : 0.901  AAD : +1.990  BAD : -1.990

Cedents:
  antecedent : Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(ATL) & destinationAirport(SFO)
  succcedent : totalFare_cat(d) 400+ USD)
  condition  : ---

Fourfold table
    |  S  |  ¬S |
----|-----|-----|
 A  |  127|   14|
----|-----|-----|
¬A  |75187|174672|
----|-----|-----|
```

- Konfidence = 127/(127+14) = = 0,901
- 90,1 % letenek splňujících Tot_Trav_Duration_cat(a) <4 h b) 4-7 h) & startingAirport(ATL) & destinationAirport(SFO) splňuje i totalFare_cat(d) 400+ USD)

KRÁTKÉ &DRAHÉ LETY– INTERPRETACE

- Např. cena více než 90 % „krátkých“ letenek z Atlanty do San Francisca přesahuje 400 USD.
- Obecně letenky z/do San Francisca bývají poměrně drahé i přes kratší délku letu. (San Francisco vystupuje v každém z výstupních pravidel, ať už jako počáteční či cílová destinace). Důvodem může být fakt, že se San Francisco nachází na západním pobřeží USA, ale většina analyzovaných destinací je blíže k pobřeží východnímu.
- Bylo by možné dále zkoumat dopady těchto vysokých cen letenek na chování zákazníků.

Total Fare Category - less than 7 hours flight



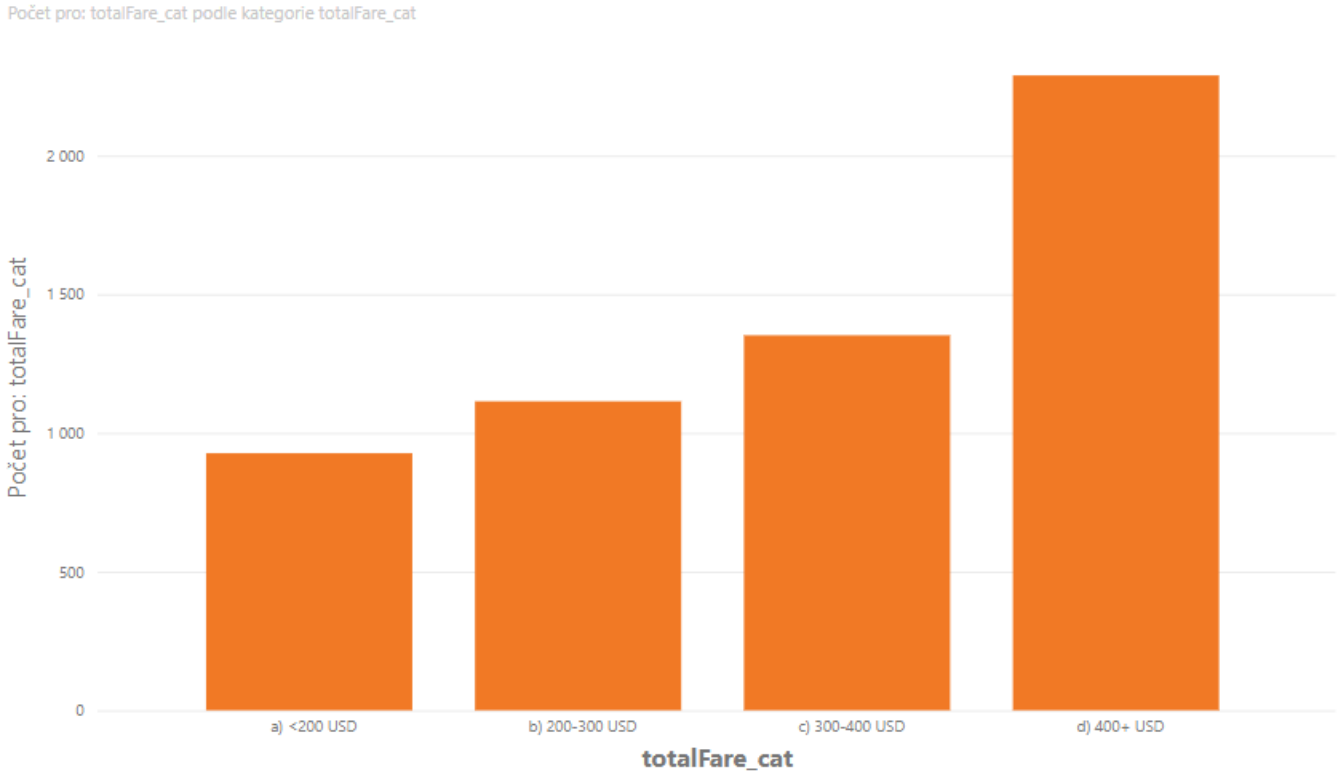
startingAirport

ATL	DFW	JFK	OAK
BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	

destinationAirport

BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	SFO
DFW	JFK	OAK	

Total Fare Category - less than 7 hours flight



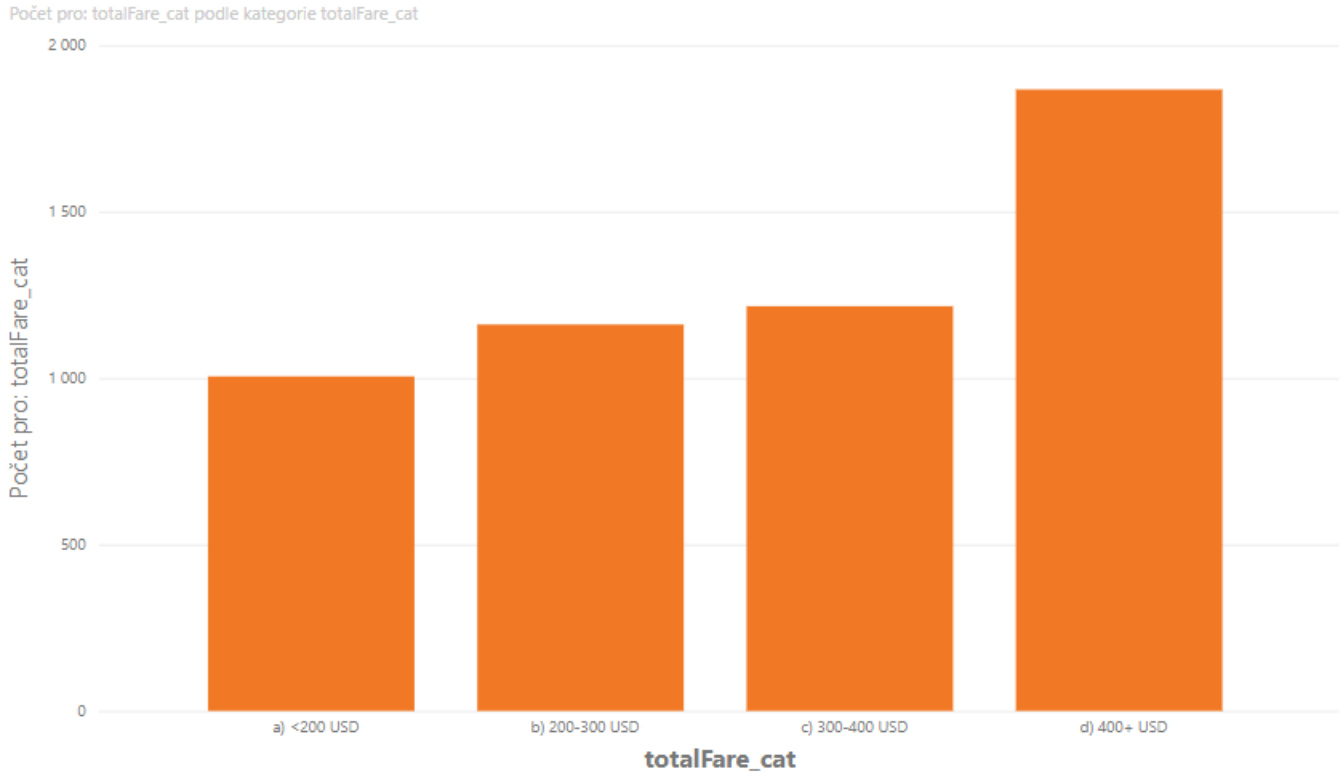
startingAirport

ATL	DFW	JFK	OAK
BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	SFO

destinationAirport

ATL	DFW	JFK	OAK
BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	

Total Fare Category - less than 7 hours flight



startingAirport

ATL	DFW	JFK	OAK
BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	

destinationAirport

ATL	DFW	JFK	OAK
BOS	DTW	LAX	ORD
CLT	EWR	LGA	PHL
DEN	IAD	MIA	SFO

IIb) DLOUHÉ & LEVNÉ LETY

- *Existují lety, které trvají naopak déle než 7 hodin, ale i přesto patří alespoň 70 % letenek na této trase do nejlevnějších cenových kategorií, tzn. jejich cena včetně všech poplatků nepřesáhne hranici 300 USD?*

- Vzhledem k povaze otázky byla pro sukcedent totalFare_cat zvolena varianta ‚lcut‘ maximální délky 2 a pro antecedent Tot_Trav_Duration_cat bude nyní brána varianta ‚rcut‘ maximální délky 2

```
clm = cleverminer(df=matice,proc='4ftMiner',
                quantifiers= {'conf':0.7, 'Base':50},
                ante ={
                     'attributes':[
                         {'name': 'Tot_Trav_Duration_cat', 'type':'rcut', 'minlen': 1, 'maxlen':2},
                         {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen':1},
                         {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                     ], 'minlen':3, 'maxlen':3, 'type':'con'},
                succ ={
                     'attributes':[
                         {'name': 'totalFare_cat', 'type':'lcut', 'minlen': 1, 'maxlen':2},
                     ], 'minlen':1, 'maxlen':1, 'type':'con'},
                )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

Task type : 4ftMiner
Number of verifications : 345
Number of rules : 45
Total time needed : 00h 00m 03s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 00s

List of rules:

RULEID	BASE	CONF	AAD	Rule
1	109	0.717	+0.479	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(ATL) & destinationAirport(ORD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
2	137	0.825	+0.702	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(BOS) & destinationAirport(EWR) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
3	64	0.711	+0.466	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(BOS) & destinationAirport(MIA) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
4	62	0.721	+0.487	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(DFW) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
5	60	0.968	+0.996	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(EWR) & destinationAirport(CLT) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
6	67	0.870	+0.794	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(LGA) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
7	53	0.841	+0.735	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(LGA) & destinationAirport(DFW) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
8	57	0.851	+0.754	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(LGA) & destinationAirport(ORD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
9	126	0.792	+0.634	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(ORD) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
10	66	0.702	+0.448	Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(PHL) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
11	162	0.794	+0.637	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(ATL) & destinationAirport(DFW) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
12	296	0.818	+0.686	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(ATL) & destinationAirport(ORD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
13	367	0.718	+0.481	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
14	770	0.772	+0.591	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(DEN) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
15	613	0.793	+0.635	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(DFW) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
16	285	0.821	+0.694	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(EWR) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
17	309	0.780	+0.609	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(MIA) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
18	200	0.741	+0.527	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(BOS) & destinationAirport(ORD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
19	113	0.796	+0.641	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(CLT) & destinationAirport(EWR) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
20	67	0.736	+0.518	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(CLT) & destinationAirport(IAD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
21	142	0.802	+0.654	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(CLT) & destinationAirport(LGA) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
22	394	0.774	+0.596	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(CLT) & destinationAirport(ORD) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
23	188	0.718	+0.480	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(DEN) & destinationAirport(DFW) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
24	177	0.760	+0.566	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(DFW) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
25	241	0.703	+0.449	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(DFW) & destinationAirport(LAX) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
26	210	0.700	+0.443	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(DFW) & destinationAirport(LGA) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
27	139	0.794	+0.638	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(EWR) & destinationAirport(ATL) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
28	194	0.773	+0.594	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(EWR) & destinationAirport(BOS) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
29	146	0.967	+0.994	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(EWR) & destinationAirport(CLT) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---
30	371	0.743	+0.533	Tot_Trav_Duration_cat(d) 10+ h c) 7-10 h) & startingAirport(EWR) & destinationAirport(DFW) => totalFare_cat(a) <200 USD b) 200-300 USD) | ---

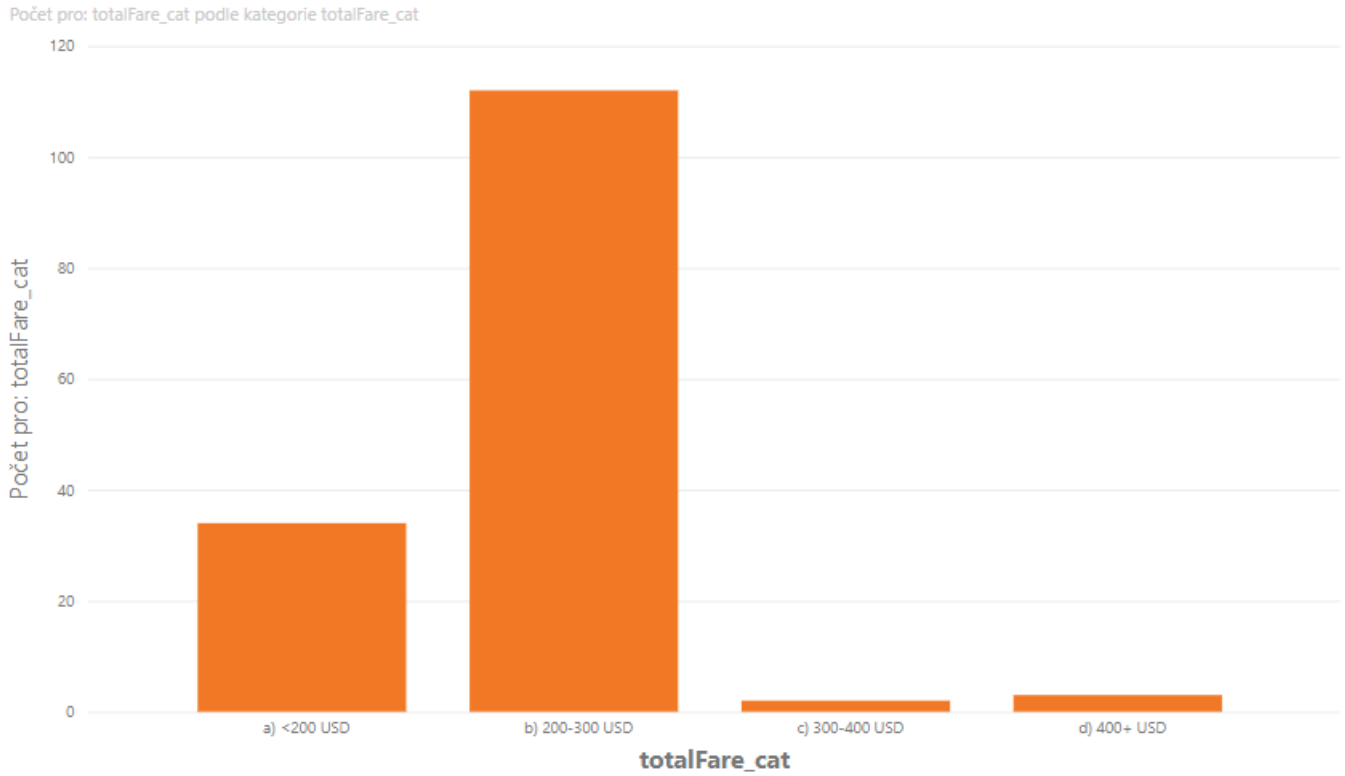
```
Rule id : 5

Base :    60  Relative base : 0.000  CONF : 0.968  AAD : +0.996  BAD : -0.996

Cedents:
  antecedent : Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(EWR) & destinationAirport(CLT)
  succcedent : totalFare_cat(a) <200 USD b) 200-300 USD)
  condition  : ---

Fourfold table
    |  S  |  ¬S |
----|-----|-----|
 A  |   60|    2|
----|-----|-----|
¬A  |121180|128758|
----|-----|-----|
```

- Konfidence = 60/(60+2) = 0,968
- 96,8 % letenek splňujících Tot_Trav_Duration_cat(d) 10+ h) & startingAirport(EWR) & destinationAirport(CLT) splňuje i totalFare_cat(a) <200 USD b) 200-300 USD)

DLOUHÉ & LEVNÉ LETY – INTERPRETACE

- Např. cena více než 96 % „dlouhých“ letenek z Newark do Charlotte nepřesahuje 300 USD.
- Obecně letenky z New Yorku (LaGuardia Airport) bývají do mnoha míst poměrně levné i přes větší délku letu.
- Bylo by možné zvážit případné zvýšení těchto cen a dále zkoumat dopady těchto zvýšených cen letenek na chování zákazníků, a tedy i na celkové zisky z daných letů.

Total Fare Category - more than 7 hours flight



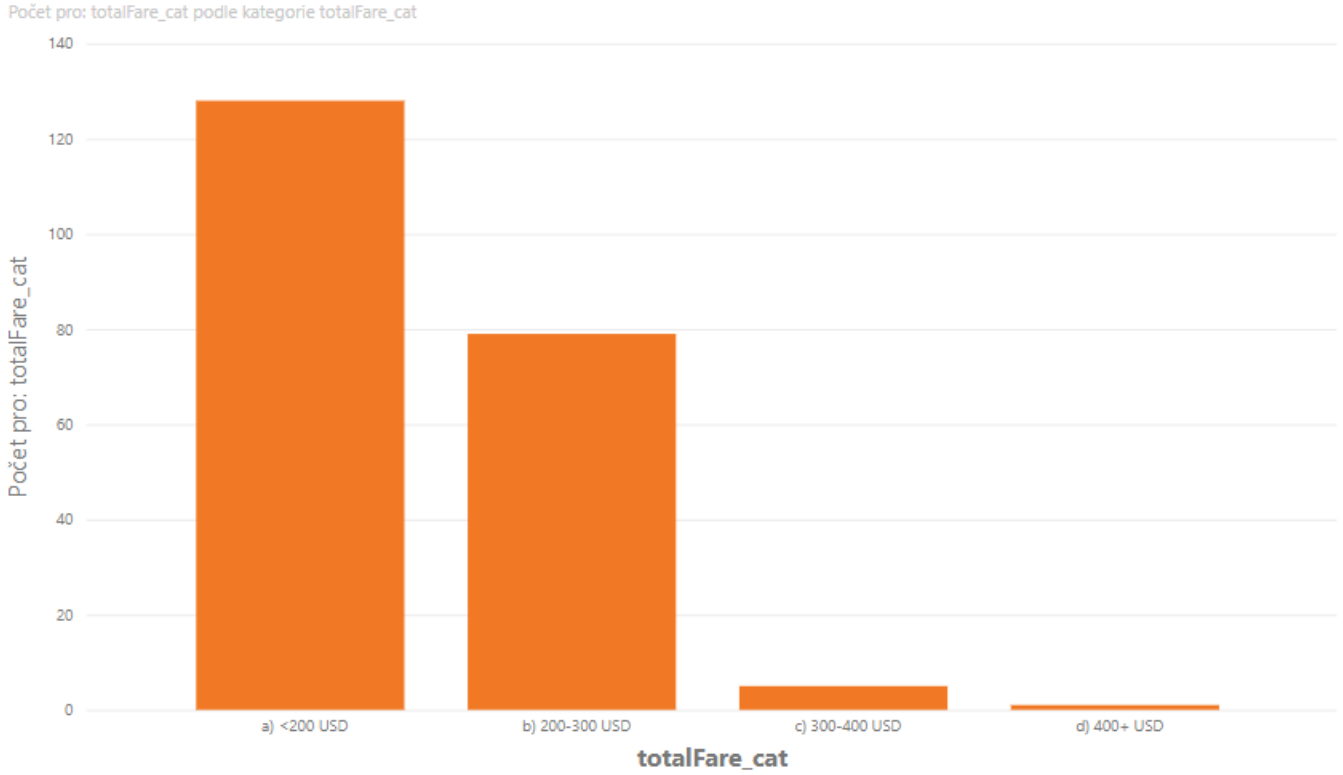
startingAirport

ATL	DTW	LAX	ORD
BOS	EWR	LGA	PHL
DEN	IAD	MIA	SFO
DFW	JFK	OAK	

destinationAirport

ATL	DFW	MIA	SFO
BOS	DTW	OAK	
CLT	IAD	ORD	
DEN	LAX	PHL	

Total Fare Category - more than 7 hours flight



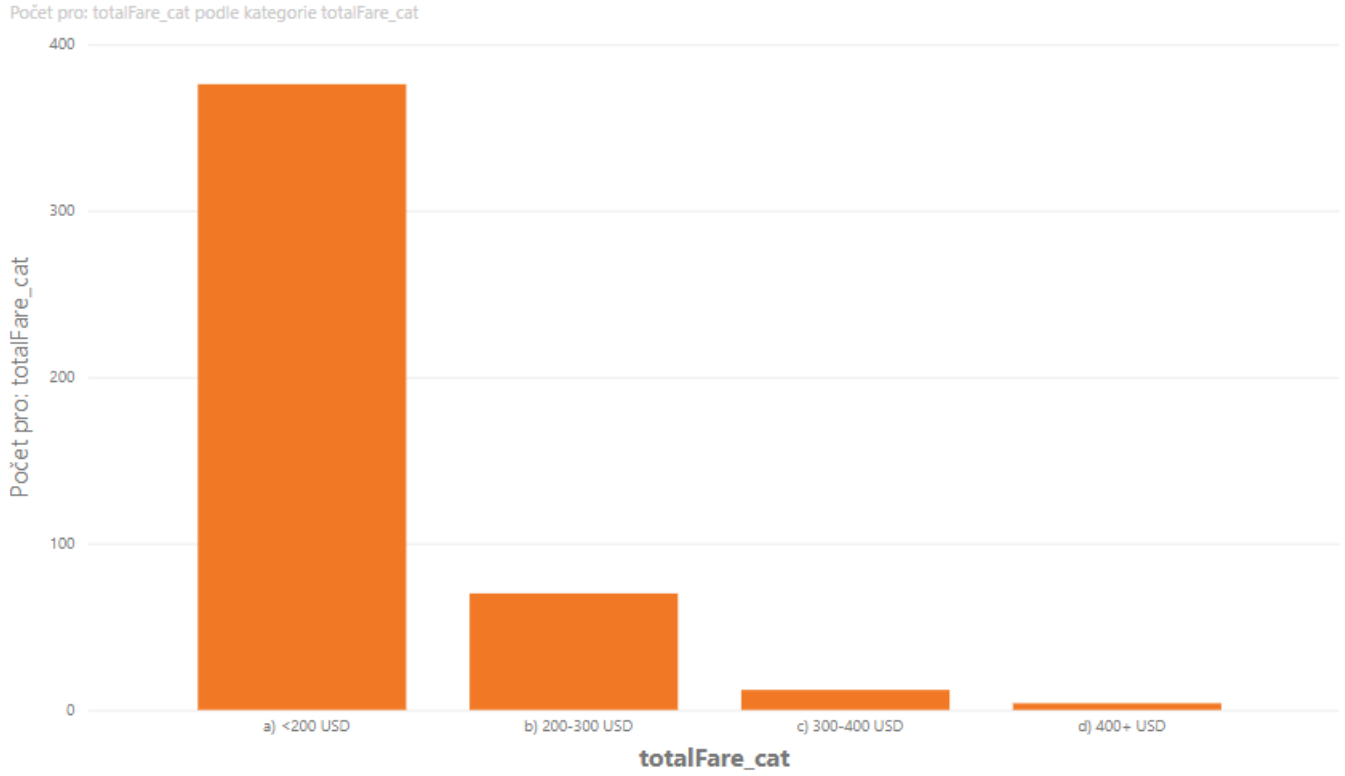
startingAirport

ATL	DFW	LAX	ORD
BOS	EWR	LGA	PHL
CLT	IAD	MIA	SFO
DEN	JFK	OAK	

destinationAirport

ATL	DFW	MIA	SFO
BOS	DTW	OAK	
CLT	IAD	ORD	
DEN	LAX	PHL	

Total Fare Category - more than 7 hours flight



startingAirport

ATL	DFW	JFK	ORD
BOS	DTW	LAX	PHL
CLT	EWR	**LGA**	SFO
DEN	IAD	OAK	

destinationAirport

ATL	DFW	**MIA**	SFO
BOS	DTW	OAK	
CLT	IAD	ORD	
DEN	LAX	PHL	

02

CF-MINER

I) CENY LETENEK & DNY V TÝDNU

- Na základě BI analýzy bylo zjištěno, nedělní a pondělní lety bývají obecně dražší, naopak letenky s odletem v úterý či středu bývají v porovnání s jinými dny výrazně levnější.
- Nyní bude cílem zjistit, zda se za nějakých podmínek ceny letenek v tyto dny významně liší.

I) CENY LETENEK & DNY V TÝDNU

- *Existuje nějaká kombinace atributů tak, že pro tuto kombinaci bude mít histogram cen letenek opačný trend, než platí obecně?*

Ia) CENY LETENEK & NEDĚLNÍ ODLETY

- *Existuje nějaká kombinace trasy, meteorologických údajů, denní doby odletu, počtu zbývajících volných míst v letadle a počtu dní do odletu tak, že pro tuto kombinaci bude histogram cen nedělních letenek klesající?*

- Vzhledem k povaze otázky byla pro atribut Weekday zvolena typ ‚one‘ s hodnotou ‚7 Sun‘, všechny ostatní atributy s volbou ‚subset‘ délky 1.

```
clm = cleverminer(df=matice,target='totalFare_cat',proc='CFMiner',
                  quantifiers= {'S_Down':3, 'Base':500},
                  cond ={
                       'attributes':[
                           {'name': 'Weekday', 'type': 'one', 'value': '7 Sun'},
                           {'name': 'dep_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'seatsRemaining', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'DaysToFlight_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'tavg_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'prcp_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'wspd_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                       ], 'minlen':2, 'maxlen':9, 'type':'con'}
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
CleverMiner task processing summary:

Task type : CFMiner
Number of verifications : 19665
Number of rules : 1839
Total time needed : 00h 00m 15s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 12s


List of rules:
RULEID BASE  S_UP  S_DOWN Condition
     1   712     0     3 Weekday(7 Sun) & DaysToFlight_cat(a) < 4 Days) & startingAirport(ORD)
     2   654     0     3 Weekday(7 Sun) & DaysToFlight_cat(a) < 4 Days) & startingAirport(ORD) & prcp_cat(a) 0-1)
     3   592     0     3 Weekday(7 Sun) & DaysToFlight_cat(d) 11+ Days) & startingAirport(BOS)
     4   801     0     3 Weekday(7 Sun) & startingAirport(LGA) & tavg_cat(d) 20-30)

Rule id : 1

Base :   712  Relative base : 0.003  Steps UP (consecutive) :     0  Steps DOWN (consecutive) :     3
6  Histogram relative maximum : 0.367 Histogram relative minimum : 0.149

Condition  : Weekday(7 Sun) & DaysToFlight_cat(a) < 4 Days) & startingAirport(ORD)

Categories in target variable  ['a) >200 USD', 'b) 200-300 USD', 'c) 300-400 USD', 'd) 400+ USD']
Histogram                      [261, 219, 126, 106]
```

CENY LETENEK & NEDĚLE – INTERPRETACE

- Celkem 4 histogramy nedělních odletů splňující zadané parametry (ale druhý z histogramů již vyplývá z prvního).
- Například nedělní letenky z Chicaga bývají neobvykle levné, zbývají-li nejvýše 3 dny do odletu.
- Naopak nedělní letenky z Bostonu lze sehnat nezvykle levně, nakupujeme-li s dostatečným předstihem.

Total Fare Category and Weekdays

DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count

Weekday	Průměr z: totalFare
7 Sun	$462
Celkem	**$462**

startingAirport

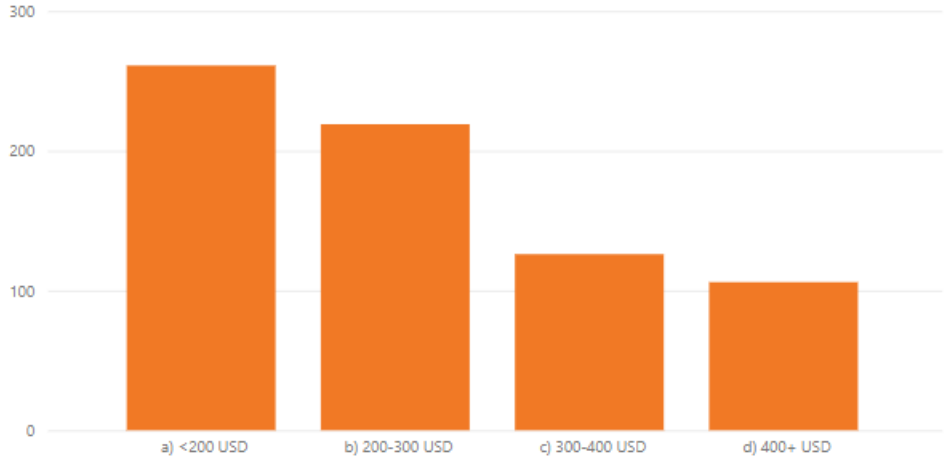
ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

totalFare_cat count



Weekday	Průměr z: totalFare
7 Sun	$286
Celkem	**$286**

Weekday

1 Mon
2 Tue
3 Wed
4 Thu
5 Fri
6 Sat
7 Sun

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

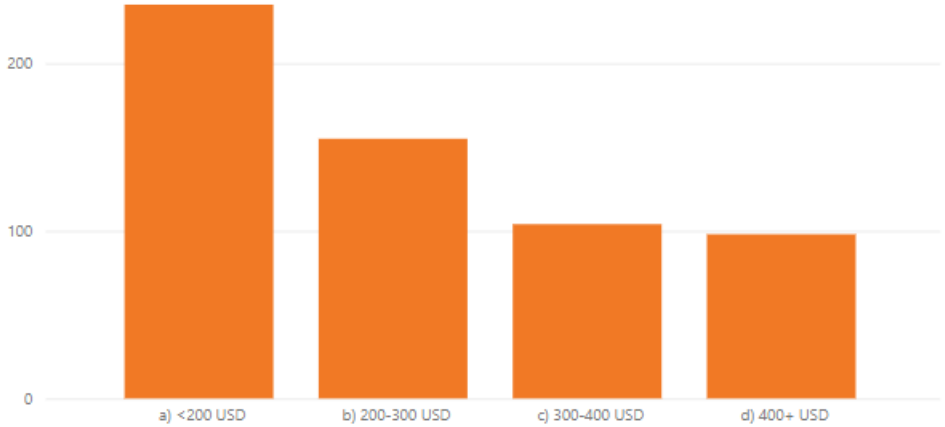
DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	**d) 11+ Days**

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- **7 Sun**

totalFare_cat count



Weekday	Průměr z: totalFare
7 Sun	$267
Celkem	**$267**

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Ib) CENY LETENEK & PONDĚLNÍ ODLETY

- *Existuje nějaká kombinace trasy, meteorologických údajů, denní doby odletu, počtu zbývajících volných míst v letadle, údajů o čekání a přestupech, informací o aerolinkách a počtu dní do odletu tak, že pro tuto kombinaci bude histogram cen pondělních letenek klesající?*

- Vzhledem k povaze otázky byla pro atribut Weekday zvolena typ ‚one‘ s hodnotou ‚1 Mon‘, všechny ostatní atributy s volbou ‚subset‘ délky 1.

```
clm = cleverminer(df=matice,target='totalFare_cat',proc='CFMiner',
                  quantifiers= {'S_Down':3, 'Base':2000},
                  cond ={
                       'attributes':[
                           {'name': 'Weekday', 'type': 'one', 'value': '1 Mon'},
                           {'name': 'dep_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'seatsRemaining', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'DaysToFlight_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'tavg_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'prcp_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'wspd_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'elapsedDays', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'NumberOfTransfers', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'Wait_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameCount', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameSummary', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                       ], 'minlen':2, 'maxlen':3, 'type':'con'}
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
Task type : CFMiner
Number of verifications : 9795
Number of rules : 1415
Total time needed : 00h 00m 09s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 07s


List of rules:
RULEID BASE  S_UP  S_DOWN Condition
     1  3189     0      3 Weekday(1 Mon) & dep_time_cat(b) Afternoon) & seatsRemaining(7)
     2  2745     0      3 Weekday(1 Mon) & dep_time_cat(b) Afternoon) & DaysToFlight_cat(d) 11+ Days)
     3  2712     0      3 Weekday(1 Mon) & dep_time_cat(b) Afternoon) & tavg_cat(d) 20-30)
     4  4417     0      3 Weekday(1 Mon) & dep_time_cat(b) Afternoon) & NumberOfTransfers(0)
     5  4417     0      3 Weekday(1 Mon) & dep_time_cat(b) Afternoon) & Wait_time_cat(a) 0 h)
     6  2066     0      3 Weekday(1 Mon) & dep_time_cat(c) Evening) & NumberOfTransfers(0)
     7  2066     0      3 Weekday(1 Mon) & dep_time_cat(c) Evening) & Wait_time_cat(a) 0 h)
     8  4432     0      3 Weekday(1 Mon) & seatsRemaining(7) & NumberOfTransfers(0)
     9  4432     0      3 Weekday(1 Mon) & seatsRemaining(7) & Wait_time_cat(a) 0 h)
    10  2474     0      3 Weekday(1 Mon) & seatsRemaining(9) & NumberOfTransfers(0)
    11  2474     0      3 Weekday(1 Mon) & seatsRemaining(9) & Wait_time_cat(a) 0 h)
    12  2231     0      3 Weekday(1 Mon) & seatsRemaining(9) & AirlineNameSummary(United)
    13  2436     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & tavg_cat(d) 20-30)
    14  2580     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & NumberOfTransfers(0)
    15  2580     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & Wait_time_cat(a) 0 h)
    16  8238     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & AirlineNameCount(1)
    17  2472     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & AirlineNameSummary(American Airlines)
    18  2000     0      3 Weekday(1 Mon) & DaysToFlight_cat(d) 11+ Days) & AirlineNameSummary(United)
    19  2153     0      3 Weekday(1 Mon) & startingAirport(BOS) & prcp_cat(a) 0-1)
    20  2991     0      3 Weekday(1 Mon) & startingAirport(BOS) & AirlineNameCount(1)
    21  2592     0      3 Weekday(1 Mon) & startingAirport(LGA) & wspd_cat(b) Light or Gentle Breeze)
    22  3135     0      3 Weekday(1 Mon) & startingAirport(LGA) & AirlineNameCount(1)
    23  2776     0      3 Weekday(1 Mon) & startingAirport(ORD)
```

CENY LETENEK & PONDĚLÍ – INTERPRETACE

- Obecně bývají nezvykle levné pondělní letenky kupované na přímé lety, ty kupované s dostatečným předstihem nebo ty kupované při větším počtu zbývajících míst.
- Také pondělní letenky z Chicaga bývají neobvykle levné, podobně jako ty nedělní.

Total Fare Category and Weekdays

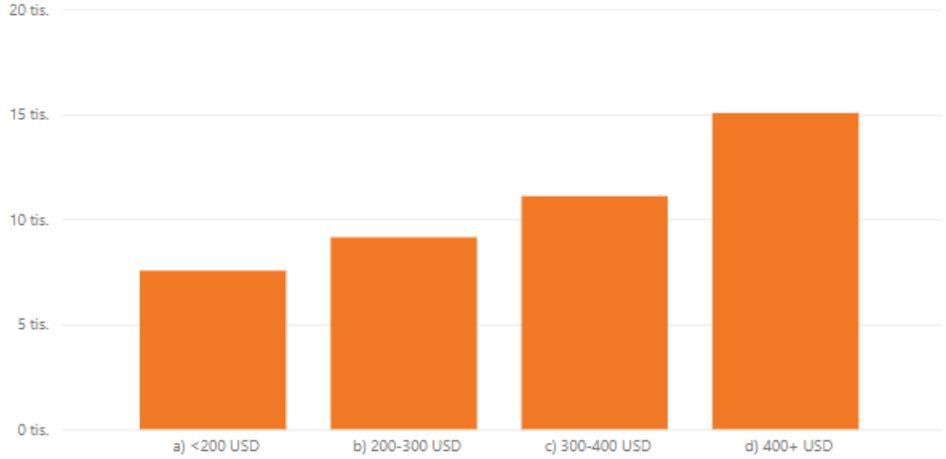
DaysToFlight_cat

a) < 4 Days	d) 11+ Days
c) 8 - 10 Days	

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
1 Mon	$374
Celkem	**$374**

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

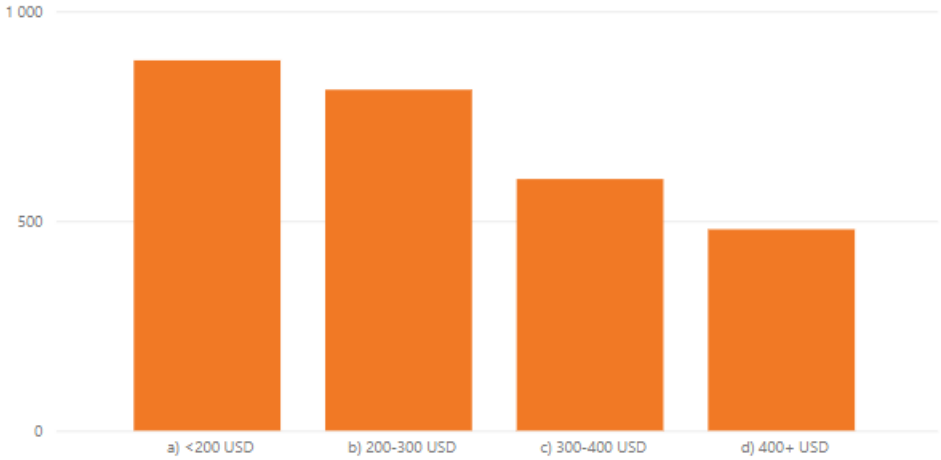
DaysToFlight_cat

a) < 4 Days	d) 11+ Days
c) 8 - 10 Days	

Weekday

1 Mon
2 Tue
3 Wed
4 Thu
5 Fri
6 Sat
7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
1 Mon	$291
Celkem	**$291**

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Ic) CENY LETENEK & ÚTERNÍ ODLETY

- *Existuje nějaká kombinace trasy, meteorologických údajů, denní doby odletu, počtu zbývajících volných míst v letadle, údajů o čekání a přestupech, informací o aerolinkách a počtu dní do odletu tak, že pro tuto kombinaci bude histogram cen úterních letenek rostoucí?*

- Vzhledem k povaze otázky byla pro atribut Weekday zvolena typ ‚one‘ s hodnotou ‚2 Tue‘, všechny ostatní atributy s volbou ‚subset‘ délky 1.

```
clm = cleverminer(df=matice,target='totalFare_cat',proc='CFMiner',
                  quantifiers= {'S_Up':3, 'Base':2000},
                  cond ={
                       'attributes':[
                           {'name': 'Weekday', 'type': 'one', 'value': '2 Tue'},
                           {'name': 'dep_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'seatsRemaining', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'DaysToFlight_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'tavg_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'prcp_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'wspd_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'elapsedDays', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'NumberOfTransfers', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'Wait_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameCount', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameSummary', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                       ], 'minlen':2, 'maxlen':4, 'type':'con'}
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
Task type : CFMiner
Number of verifications : 27813
Number of rules : 5518
Total time needed : 00h 00m 32s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 29s


List of rules:
RULEID BASE  S_UP  S_DOWN Condition
     1  2062     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & seatsRemaining(2)
     2  2053     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & seatsRemaining(2) & elapsedDays(0)
     3  2127     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & tavg_cat(c) 10-20) & AirlineNameCount(2)
     4  2211     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & prcp_cat(a) 0-1) & AirlineNameCount(2)
     5  2348     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & wspd_cat(b) Light or Gentle Breeze) & AirlineNameCount(2)
     6  2111     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & elapsedDays(0) & NumberOfTransfers(2)
     7  2966     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & elapsedDays(0) & AirlineNameCount(2)
     8  2238     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & NumberOfTransfers(2)
     9  2167     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & NumberOfTransfers(2) & Wait_time_cat(c) 2+ h)
    10  2637     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & Wait_time_cat(c) 2+ h) & AirlineNameCount(2)
    11  3060     3     0 Weekday(2 Tue) & dep_time_cat(a) Morning) & AirlineNameCount(2)
    12  2832     3     0 Weekday(2 Tue) & dep_time_cat(c) Evening) & DaysToFlight_cat(a) < 4 Days)
    13  2236     3     0 Weekday(2 Tue) & dep_time_cat(c) Evening) & DaysToFlight_cat(a) < 4 Days) & prcp_cat(a) 0-1)
    14  2153     3     0 Weekday(2 Tue) & dep_time_cat(c) Evening) & prcp_cat(a) 0-1) & Wait_time_cat(c) 2+ h)
    15  2265     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & prcp_cat(a) 0-1) & elapsedDays(1)
    16  4898     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & wspd_cat(b) Light or Gentle Breeze) & Wait_time_cat(c) 2+ h)
    17  2080     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & wspd_cat(b) Light or Gentle Breeze) & AirlineNameSummary(Delta)
    18  2527     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & elapsedDays(1)
    19  4050     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & AirlineNameCount(1) & AirlineNameSummary(Delta)
    20  4050     3     0 Weekday(2 Tue) & DaysToFlight_cat(a) < 4 Days) & AirlineNameSummary(Delta)
    21  2059     3     0 Weekday(2 Tue) & DaysToFlight_cat(c) 8 - 10 Days) & AirlineNameCount(2)
    22  2191     3     0 Weekday(2 Tue) & startingAirport(SFO) & prcp_cat(a) 0-1)
    23  2238     3     0 Weekday(2 Tue) & startingAirport(SFO) & wspd_cat(b) Light or Gentle Breeze)
    24  2014     3     0 Weekday(2 Tue) & destinationAirport(OAK)
```

CENY LETENEK & ÚTERÝ – INTERPRETACE

- Například lety z Oaklandu bývají na průměrné úterní ceny nezvykle drahé.
- Dále jsou dražší úterní letenky od letecké společnosti Delta
- Další zajímavé nalezené pravidlo říká, že v úterý bývají drahé také večerní letenky kupované maximálně 3 dny před odletem.

Total Fare Category and Weekdays

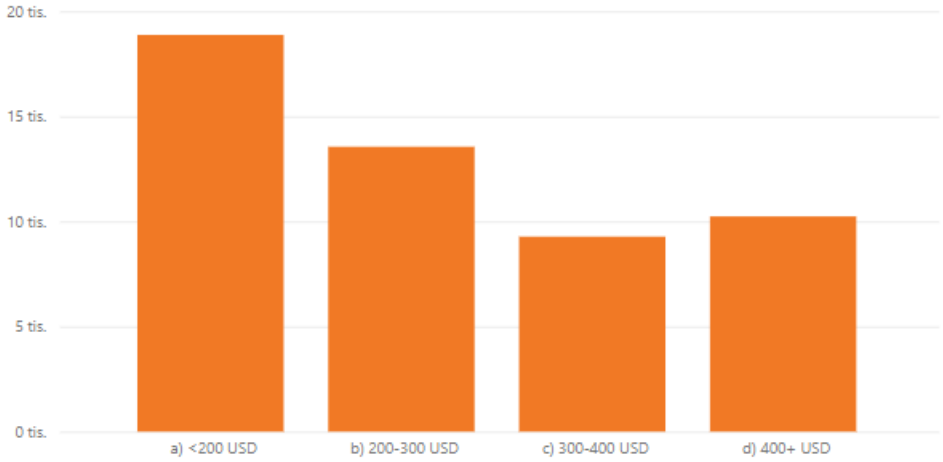
DaysToFlight_cat

- a) < 4 Days
- d) 11+ Days
- c) 8 - 10 Days

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
2 Tue	$290
Celkem	**$290**

dep_time_cat

- [ ] a) Morning
- [ ] b) Afternoon
- [ ] c) Evening
- [ ] d) Night

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

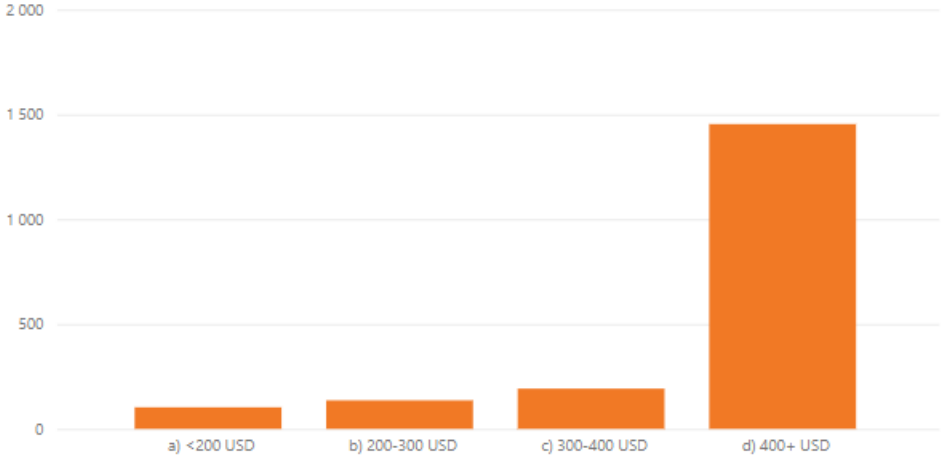
DaysToFlight_cat

- a) < 4 Days
- d) 11+ Days
- c) 8 - 10 Days

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
2 Tue	$506
Celkem	**$506**

dep_time_cat

- a) Morning
- b) Afternoon
- c) Evening
- d) Night

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

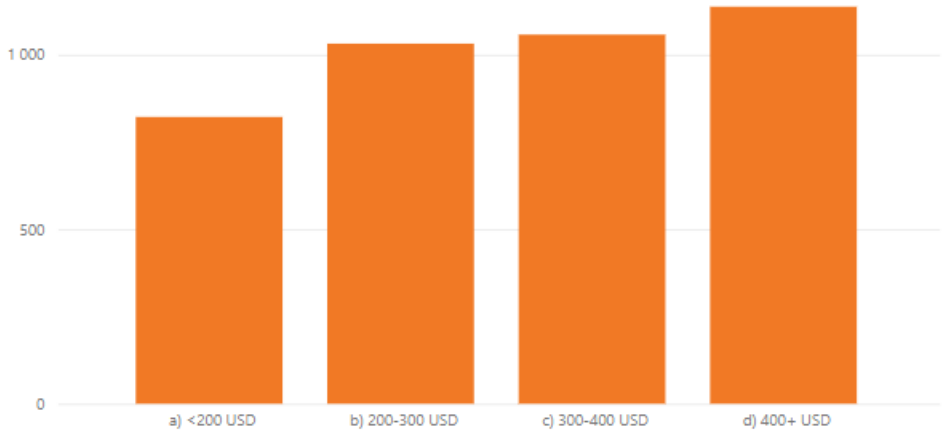
DaysToFlight_cat

a) < 4 Days	d) 11+ Days
c) 8 - 10 Days	

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
2 Tue	$331
Celkem	**$331**

AirlineNameSummary

- Cape Air|United
- Delta
- Delta|Alaska Airlines

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays

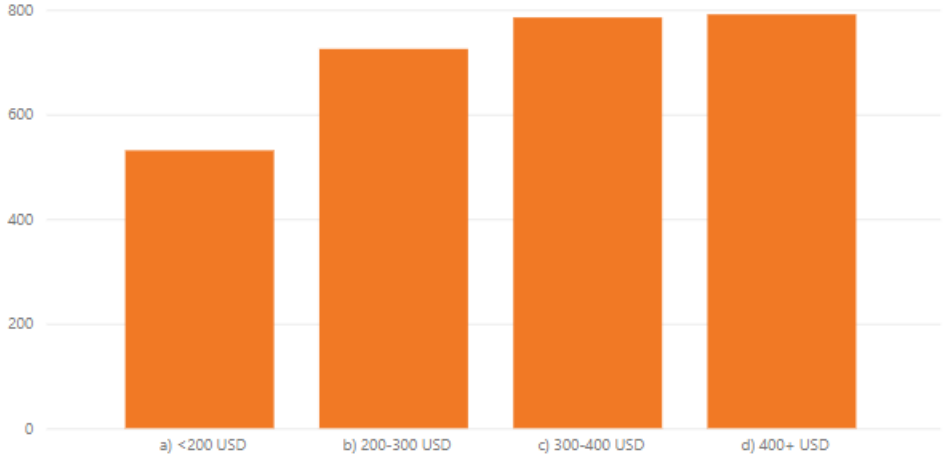
DaysToFlight_cat

- a) < 4 Days
- d) 11+ Days
- c) 8 - 10 Days

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

totalFare_cat count



Weekday	Průměr z: totalFare
2 Tue	$362
Celkem	**$362**

dep_time_cat

- a) Morning
- b) Afternoon
- c) Evening

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Id) CENY LETENEK & STŘEDEČNÍ ODLETY

- *Existuje nějaká kombinace trasy, meteorologických údajů, denní doby odletu, počtu zbývajících volných míst v letadle, údajů o čekání a přestupech, informací o aerolinkách a počtu dní do odletu tak, že pro tuto kombinaci bude histogram cen středečních letenek rostoucí?*

- Vzhledem k povaze otázky byla pro atribut Weekday zvolena typ ‚one‘ s hodnotou ‚3 Wed‘, všechny ostatní atributy s volbou ‚subset‘ délky 1.

```
clm = cleverminer(df=matice,target='totalFare_cat',proc='CFMiner',
                  quantifiers= {'S_Up':3, 'Base':2000},
                  cond ={
                       'attributes':[
                           {'name': 'Weekday', 'type': 'one', 'value': '2 Tue'},
                           {'name': 'dep_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'seatsRemaining', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'DaysToFlight_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'startingAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'destinationAirport', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'tavg_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'prcp_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'wspd_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'elapsedDays', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'NumberOfTransfers', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'Wait_time_cat', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameCount', 'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'AirlineNameSummary', 'type': 'subset', 'minlen': 1, 'maxlen': 1}
                       ], 'minlen':2, 'maxlen':4, 'type':'con'}
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
Task type : CFMiner
Number of verifications : 27408
Number of rules : 5509
Total time needed : 00h 00m 31s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 29s


List of rules:
RULEID BASE  S_UP  S_DOWN Condition
     1  2344     3     0 Weekday(3 Wed) & dep_time_cat(a) Morning) & DaysToFlight_cat(a) < 4 Days) & tavg_cat(c) 10-20)
     2  2366     3     0 Weekday(3 Wed) & dep_time_cat(a) Morning) & DaysToFlight_cat(b) 4 - 7 Days) & tavg_cat(c) 10-20)
     3  2430     3     0 Weekday(3 Wed) & dep_time_cat(a) Morning) & elapsedDays(0) & AirlineNameCount(2)
     4  2149     3     0 Weekday(3 Wed) & dep_time_cat(a) Morning) & Wait_time_cat(c) 2+ h) & AirlineNameCount(2)
     5  2491     3     0 Weekday(3 Wed) & dep_time_cat(a) Morning) & AirlineNameCount(2)
     6  3451     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & tavg_cat(c) 10-20)
     7  2148     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & tavg_cat(c) 10-20) & elapsedDays(1)
     8  2546     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & prcp_cat(a) 0-1) & elapsedDays(1)
     9  2394     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & wspd_cat(b) Light or Gentle Breeze) & elapsedDays(1)
    10  2212     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & wspd_cat(b) Light or Gentle Breeze) & NumberOfTransfers(1)
    11  3492     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & elapsedDays(1)
    12  2612     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & elapsedDays(1) & NumberOfTransfers(1)
    13  2116     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & elapsedDays(1) & Wait_time_cat(c) 2+ h)
    14  3262     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & NumberOfTransfers(1)
    15  2179     3     0 Weekday(3 Wed) & dep_time_cat(c) Evening) & Wait_time_cat(c) 2+ h)
    16  2508     3     0 Weekday(3 Wed) & DaysToFlight_cat(a) < 4 Days) & tavg_cat(c) 10-20) & Wait_time_cat(c) 2+ h)
    17  2047     3     0 Weekday(3 Wed) & DaysToFlight_cat(a) < 4 Days) & prcp_cat(b) 1-10)
    18  2530     3     0 Weekday(3 Wed) & DaysToFlight_cat(b) 4 - 7 Days) & tavg_cat(c) 10-20) & Wait_time_cat(c) 2+ h)
    19  2074     3     0 Weekday(3 Wed) & DaysToFlight_cat(b) 4 - 7 Days) & prcp_cat(b) 1-10)
    20  2251     3     0 Weekday(3 Wed) & DaysToFlight_cat(b) 4 - 7 Days) & wspd_cat(b) Light or Gentle Breeze) & Wait_time_cat(c) 2+ h)
    21  2504     3     0 Weekday(3 Wed) & startingAirport(SFO)
    22  2082     3     0 Weekday(3 Wed) & destinationAirport(PHL)
```

CENY LETENEK & STŘEDA – INTERPRETACE

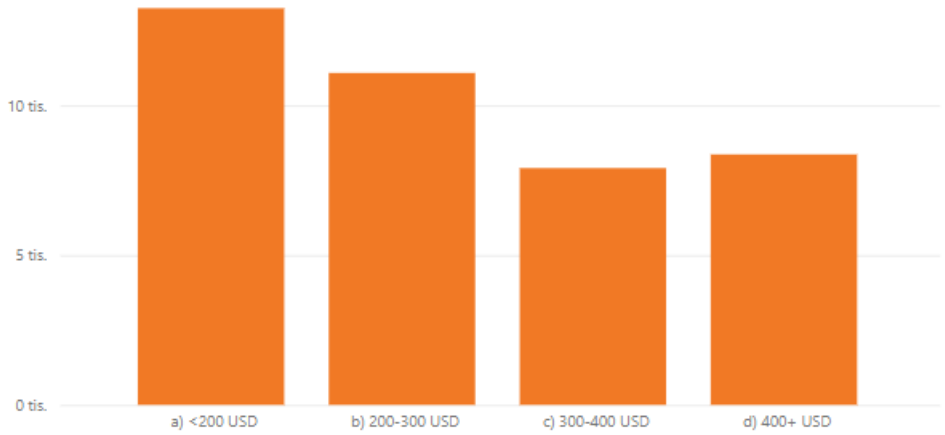
- Středeční lety ze San Francisca nebo středeční lety do Philadeplhie jsou dražší než jiné.
- To platí také například s cestami, které mají plánovaný přílet do cílové destinace až následující den, tedy ve čtvrtek.
- Zajímavá se mohou zdát také nalezená pravidla zahrnující údaje o počasí. Dražší byly například středeční letenky nakoupené maximálně týden předem do míst, kde byl denní úhrn srážek v den příletu 1-10 mm, nebo ty nakoupené maximálně týden předem s více než dvouhodinovým čekáním do míst, kde se teplota pohybovala kolem 10-20 °C.

Total Fare Category and Weekdays

DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

totalFare_cat count



Weekday	Průměr z: totalFare
3 Wed	$299
Celkem	**$299**

elapsedDays

0	1

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

Total Fare Category and Weekdays

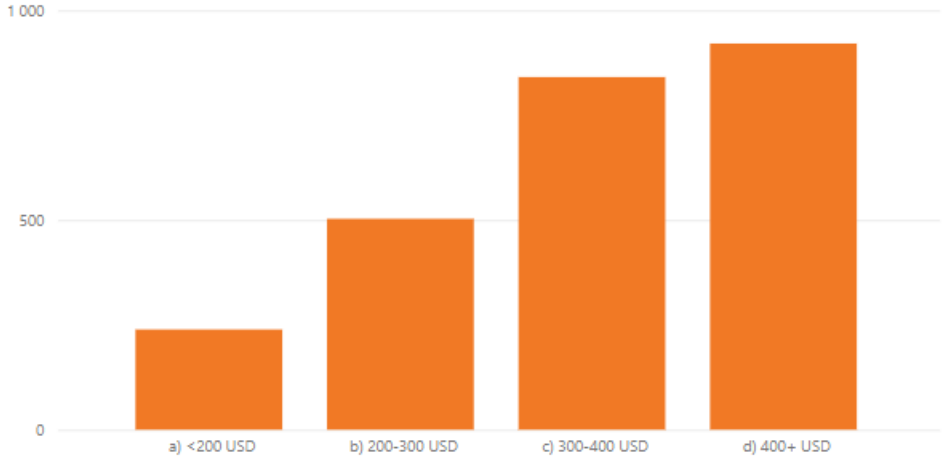
Weekday

1 Mon
2 Tue
3 Wed
4 Thu
5 Fri
6 Sat
7 Sun

DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

totalFare_cat count



Weekday	Průměr z: totalFare
3 Wed	$402
Celkem	**$402**

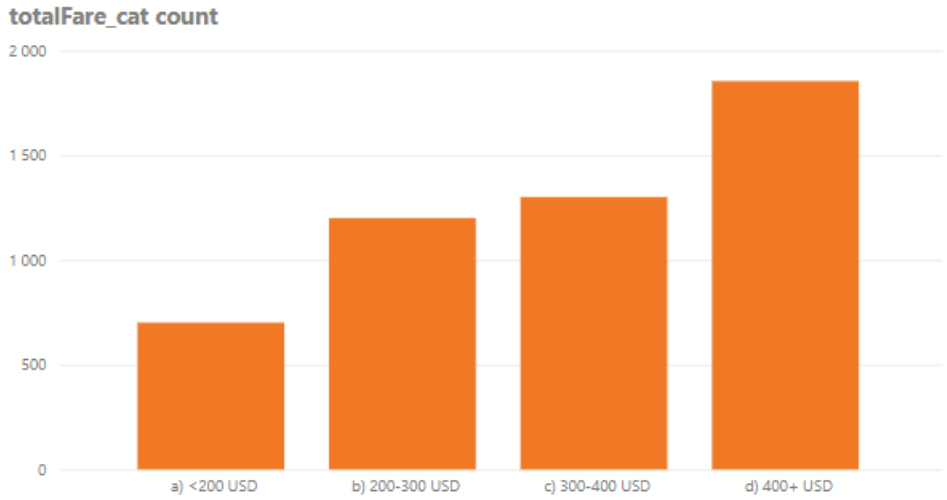
dep_time_cat

a) Morning	b) Afternoon	c) Evening	d) Night

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Total Fare Category and Weekdays



DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

Weekday	Průměr z: totalFare
3 Wed	$391
Celkem	**$391**

elapsedDays

0	1

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

Weekday

- 1 Mon
- 2 Tue
- 3 Wed
- 4 Thu
- 5 Fri
- 6 Sat
- 7 Sun

Total Fare Category and Weekdays

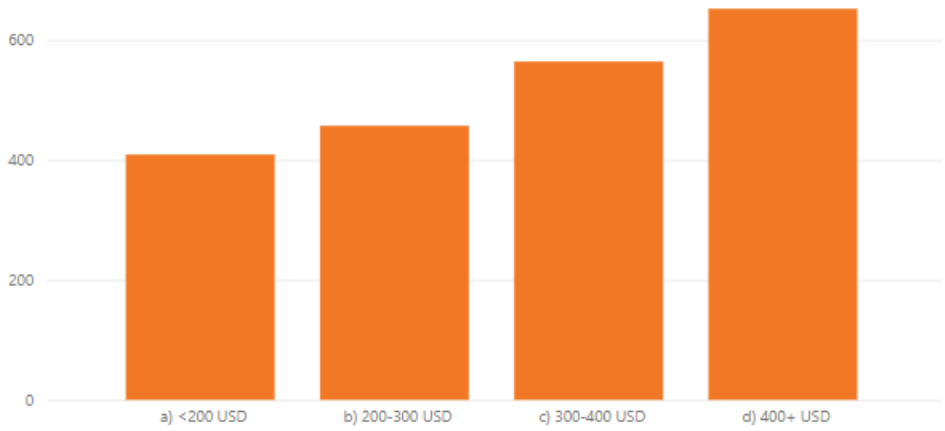
Weekday

1 Mon
2 Tue
3 Wed
4 Thu
5 Fri
6 Sat
7 Sun

DaysToFlight_cat

a) < 4 Days	c) 8 - 10 Days
b) 4 - 7 Days	d) 11+ Days

totalFare_cat count



Weekday	Průměr z: totalFare
3 Wed	$342
Celkem	**$342**

destinationAirport

LAX | LGA | MIA | OAK | ORD | PHL

startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	SFO
BOS	DEN	DTW	IAD	LAX	MIA	ORD	

CF-MINER – ZÁVĚR

- Je vcelku logické, že ceny letenek na konci víkendu bývají vyšší, uprostřed týdne naopak najdeme letenky levnější. Cílem bylo nalézt podmínky, při kterých tato všeobecně známá pravidla neplatí.
- Ceny některých letenek jsou v určité dny možná podhodnoceny. Bylo by možné zvážit případné zvýšení těchto cen a dále zkoumat dopady těchto zvýšených cen letenek na chování zákazníků, a tedy i na celkové zisky z daných letů.
- Zároveň jsou ale také dny, u kterých jsou ceny letenek za určitých podmínek neobvykle drahé. Bylo by možné dále zkoumat dopady těchto vysokých cen letenek na chování zákazníků.

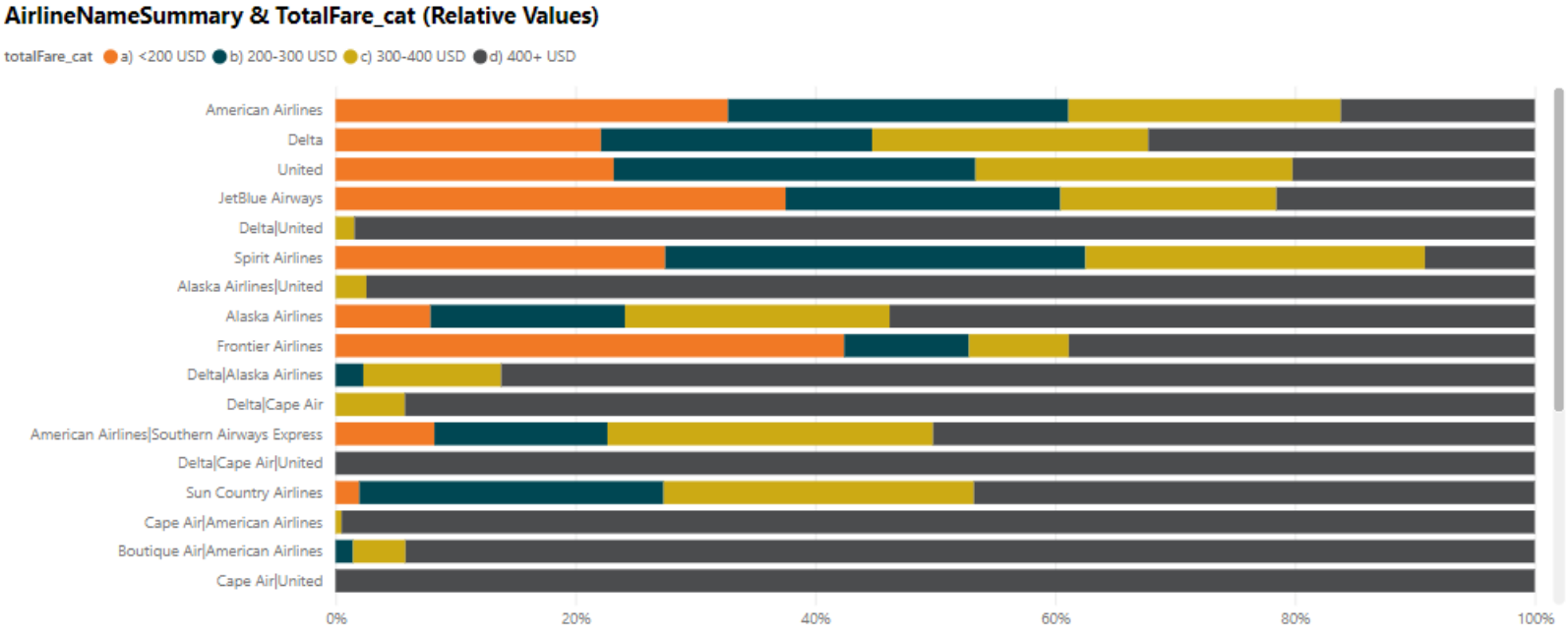
03

SD4FT-MINER

I) CENY LETENEK & AEROLINKY

- Následující otázky vychází z BI analýzy, která za obecně levnější aerolinku označila například American Airlines, za velmi drahou v porovnání s ostatními zase jednoznačně Alaska Airlines. Cílem je proto najít takové trasy, na kterých lze létat s některými aerolinkami levněji než s American Airlines, resp. dráž než s Alaska Airlines.

Airlines and Flight Prices



startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	PHL
BOS	DEN	DTW	IAD	LAX	MIA	ORD	SFO

destinationAirport

- ATL
- BOS
- CLT
- DEN
- DFW
- DTW
- EWR
- IAD
- JFK
- LAX
- LGA
- MIA
- OAK
- ORD
- PHL

Ia) CENY LETENEK & AEROLINKY

- *Je pro některé trasy relativní četnost levných letenek (<200 USD) více než 1.5x menší pro American Airlines oproti některé z jiných aerolinek (resp. jiné skupiny aerolinek)?*

- Vzhledem k povaze otázky byla pro atribut totalFare_cat zvolen typ ‚one‘ s hodnotou ‚a) <200 USD‘, a pro druhou submatici u atributu ‚AirlineNameSummary‘ typ ‚one‘ s hodnotou ‚American Airlines‘, všechny ostatní atributy s volbou ‚subset‘ délky 1.

```
clm = cleverminer(df=matice,proc='SD4ftMiner',
                  quantifiers= {'Base1':50, 'Base2':50, 'Ratioconf' : 1.5},
                  ante ={
                       'attributes':[
                           {'name': 'destinationAirport',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                           {'name': 'startingAirport',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                       ], 'minlen':2, 'maxlen':2, 'type':'con'},
                  succ ={
                       'attributes':[
                           {'name': 'totalFare_cat', 'type': 'one', 'value':'a) <200 USD'}
                       ], 'minlen':1, 'maxlen':1, 'type':'con'},
                  frst ={
                       'attributes':[
                           {'name': 'AirlineNameSummary',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                       ], 'minlen':1, 'maxlen':1, 'type':'con'},
                  scnd ={
                       'attributes':[
                           {'name': 'AirlineNameSummary',  'type': 'one', 'value': 'American Airlines'},
                       ], 'minlen':1, 'maxlen':1, 'type':'con'},
                  )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

```
Number of verifications : 4959
Number of rules : 18
Total time needed : 00h 00m 08s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 05s


List of rules:
RULEID BASE1 BASE2 RatioConf DeltaConf Rule
     1   133    50    2.010    +0.292  destinationAirport(LAX) & startingAirport(SFO) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Alaska Airlines) x AirlineNameSummary(American Airlines)
     2   190   103    1.585    +0.206  destinationAirport(ATL) & startingAirport(DFW) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(American Airlines)
     3    63    62    1.746    +0.131  destinationAirport(DEN) & startingAirport(ATL) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(American Airlines)
     4   106    50    3.332    +0.248  destinationAirport(DFW) & startingAirport(ORD) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(American Airlines)
     5    64    61    1.742    +0.257  destinationAirport(IAD) & startingAirport(BOS) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(American Airlines)
     6   120    50    2.129    +0.326  destinationAirport(LAX) & startingAirport(SFO) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(American Airlines)
     7    87   134    2.086    +0.401  destinationAirport(ATL) & startingAirport(PHL) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Frontier Airlines) x AirlineNameSummary(American Airlines)
     8    52    62    1.865    +0.185  destinationAirport(EWR) & startingAirport(MIA) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(JetBlue Airways) x AirlineNameSummary(American Airlines)
     9   111   101    1.632    +0.094  destinationAirport(LAX) & startingAirport(JFK) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(JetBlue Airways) x AirlineNameSummary(American Airlines)
    10    98   134    1.782    +0.289  destinationAirport(ATL) & startingAirport(PHL) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Spirit Airlines) x AirlineNameSummary(American Airlines)
    11    93   113    1.590    +0.233  destinationAirport(DFW) & startingAirport(ATL) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Spirit Airlines) x AirlineNameSummary(American Airlines)
    12    66   121    1.838    +0.157  destinationAirport(DFW) & startingAirport(LAX) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Spirit Airlines) x AirlineNameSummary(American Airlines)
    13    58    50    3.188    +0.233  destinationAirport(DFW) & startingAirport(ORD) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Spirit Airlines) x AirlineNameSummary(American Airlines)
    14    83    98    1.748    +0.254  destinationAirport(DTW) & startingAirport(ATL) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(Spirit Airlines) x AirlineNameSummary(American Airlines)
    15   173   137    1.938    +0.367  destinationAirport(DFW) & startingAirport(EWR) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(United) x AirlineNameSummary(American Airlines)
    16    73    50    2.932    +0.206  destinationAirport(DFW) & startingAirport(ORD) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(United) x AirlineNameSummary(American Airlines)
    17   141    61    1.858    +0.297  destinationAirport(IAD) & startingAirport(BOS) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(United) x AirlineNameSummary(American Airlines)
    18   230    50    2.689    +0.488  destinationAirport(LAX) & startingAirport(SFO) => totalFare_cat(a) >200 USD) | --- : AirlineNameSummary(United) x AirlineNameSummary(American Airlines)


Rule id : 1

Base1 :   133 Base2 :    50  Relative base 1 : 0.001 Relative base 2 : 0.000 CONF1 : 0.581  CONF2 : +0.289  Delta Conf : +0.292 Ratio Conf : +2.010

Cedents:
  antecedent : destinationAirport(LAX) & startingAirport(SFO)
  succcedent : totalFare_cat(a) >200 USD)
  condition  : ---
  first set  : AirlineNameSummary(Alaska Airlines)
  second set : AirlineNameSummary(American Airlines)

Fourfold tables:
FRST|  S  |  ¬S |  SCND|  S  |  ¬S |
----|-----|-----|  ----|-----|-----|
 A  |  133|   96|   A  |   50|  123|
----|-----|-----|  ----|-----|-----|
¬A  |  446| 6632|  ¬A  |25084|51540|
```

- $\textbf{RatioConf} = \frac{133/(133+96)}{50/(50+123)} = \frac{0{,}581}{0{,}289} = \mathbf{2{,}010}$
- Relativní četnost letenek ze SFO do LAX, které stojí méně než 200 USD, je u Alaska Airlines 2,01x vyšší než u American Airlines.

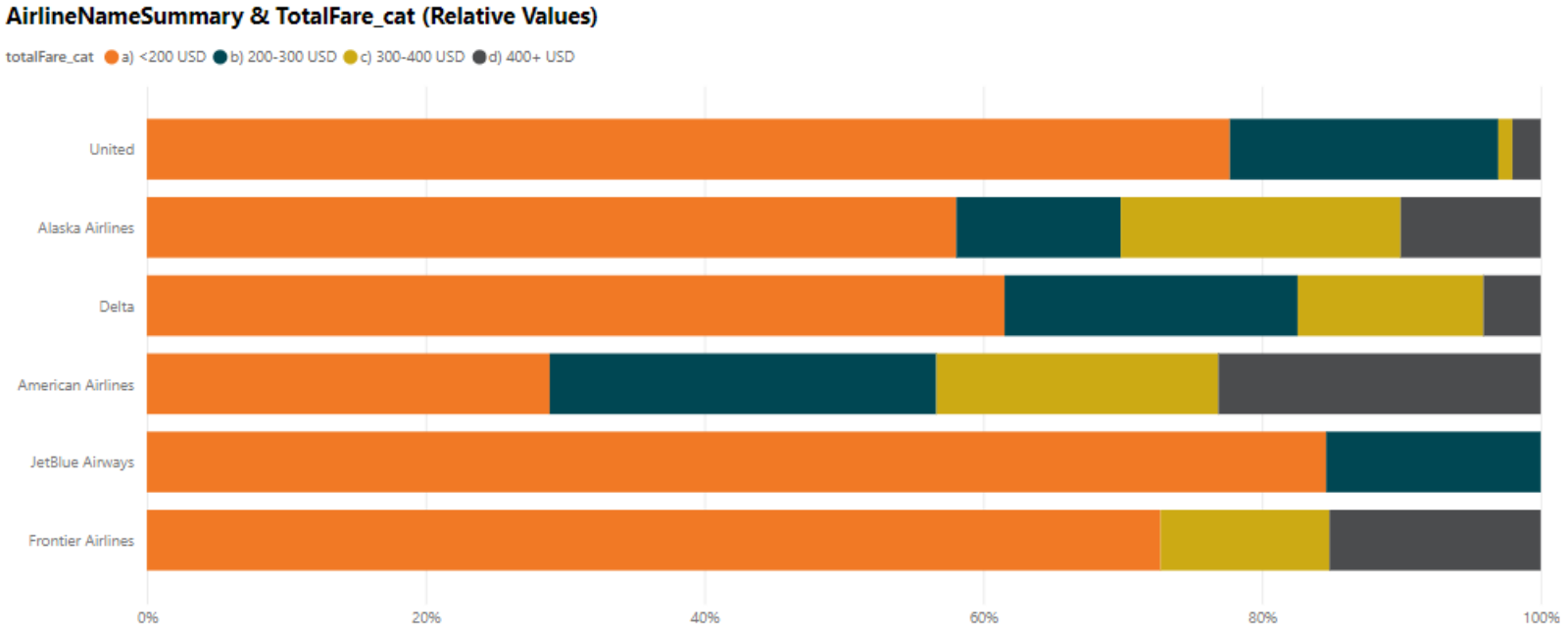
CENY LETENEK & AEROLINKY – INTERPRETACE I

- Především se to týká letenek ze San Franciska do Los Angeles nebo letenek z Dallasu do několika míst.
- Relativní četnost levných letenek mezi těmi z Dallasu do Chicaga je v případě aerolinek Delta či United přibližně 3x vyšší než relativní četnost levných letenek mezi lety z Dallasu do Chicaga se společností American Airlines.

CENY LETENEK & AEROLINKY – INTERPRETACE II

- Obzvlášť zajímavé je pak také první z nalezených pravidel, a to sice že relativní četnost levných letenek mezi těmi ze San Franciska do Los Angeles je v případě aerolinek Alaska Airlines (tzn. letecké společnosti, která se obvykle řadí mezi nejdražší na trhu, jak vyplývá z předchozí analýzy) přibližně 2x vyšší než relativní četnost levných letenek mezi lety ze San Franciska do Los Angeles se společností American Airlines.

Airlines and Flight Prices



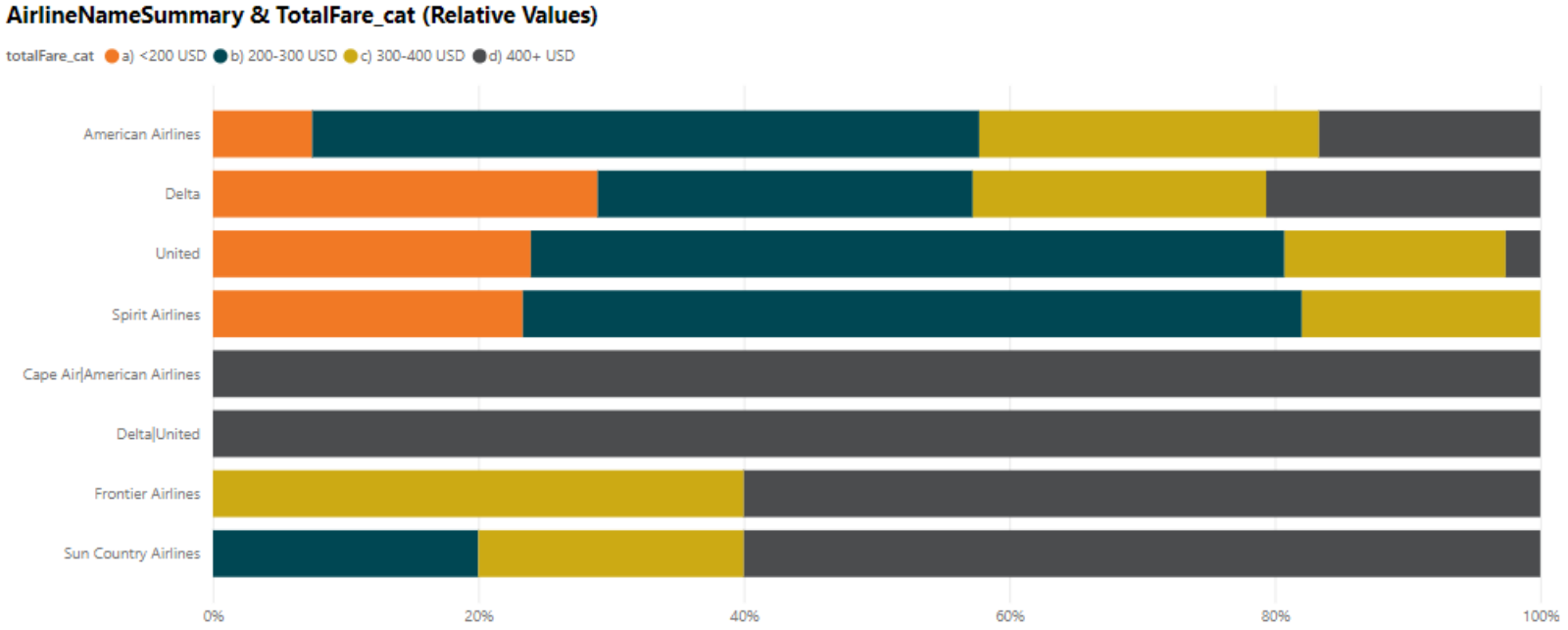
startingAirport

ATL	CLT	DFW	EWR	JFK	MIA	ORD	SFO
BOS	DEN	DTW	IAD	LGA	OAK	PHL	

destinationAirport

- ATL
- BOS
- CLT
- DEN
- DFW
- DTW
- EWR
- IAD
- JFK
- LAX
- LGA
- MIA
- OAK
- ORD
- PHL

Airlines and Flight Prices



startingAirport

ATL	CLT	DFW	EWR	JFK	LGA	OAK	SFO
BOS	DEN	DTW	IAD	LAX	MIA	PHL	

destinationAirport

ATL
BOS
CLT
DEN
DTW
EWR
IAD
JFK
LAX
LGA
MIA
OAK
ORD
PHL
SFO

Ib) CENY LETENEK & AEROLINKY

- *Je pro některé trasy relativní četnost drahých letenek (400+ USD) více než 1.5x menší pro Alaska Airlines oproti některé z jiných aerolinek (resp. jiné skupiny aerolinek)?*

- Vzhledem k povaze otázky byla pro atribut totalFare_cat zvolen typ ‚one' s hodnotou ‚d) 400+ USD', a pro druhou submatici u atributu ‚AirlineNameSummary' typ ‚one' s hodnotou ‚Alaska Airlines', všechny ostatní atributy s volbou ‚subset' délky 1.

```
clm = cleverminer(df=matice,proc='SD4ftMiner',
              quantifiers= {'Base1':50, 'Base2':50, 'Ratioconf' : 1.5},
              ante ={
                   'attributes':[
                       {'name': 'destinationAirport',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                       {'name': 'startingAirport',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                   ], 'minlen':2, 'maxlen':2, 'type':'con'},
              succ ={
                   'attributes':[
                       {'name': 'totalFare_cat', 'type': 'one', 'value':'d) 400+ USD'}
                   ], 'minlen':1, 'maxlen':1, 'type':'con'},
              frst ={
                   'attributes':[
                       {'name': 'AirlineNameSummary',  'type': 'subset', 'minlen': 1, 'maxlen': 1},
                   ], 'minlen':1, 'maxlen':1, 'type':'con'},
              scnd ={
                   'attributes':[
                       {'name': 'AirlineNameSummary',  'type': 'one', 'value': 'Alaska Airlines'},
                   ], 'minlen':1, 'maxlen':1, 'type':'con'},
              )


clm.print_summary()
clm.print_rulelist()
clm.print_rule(1)
```

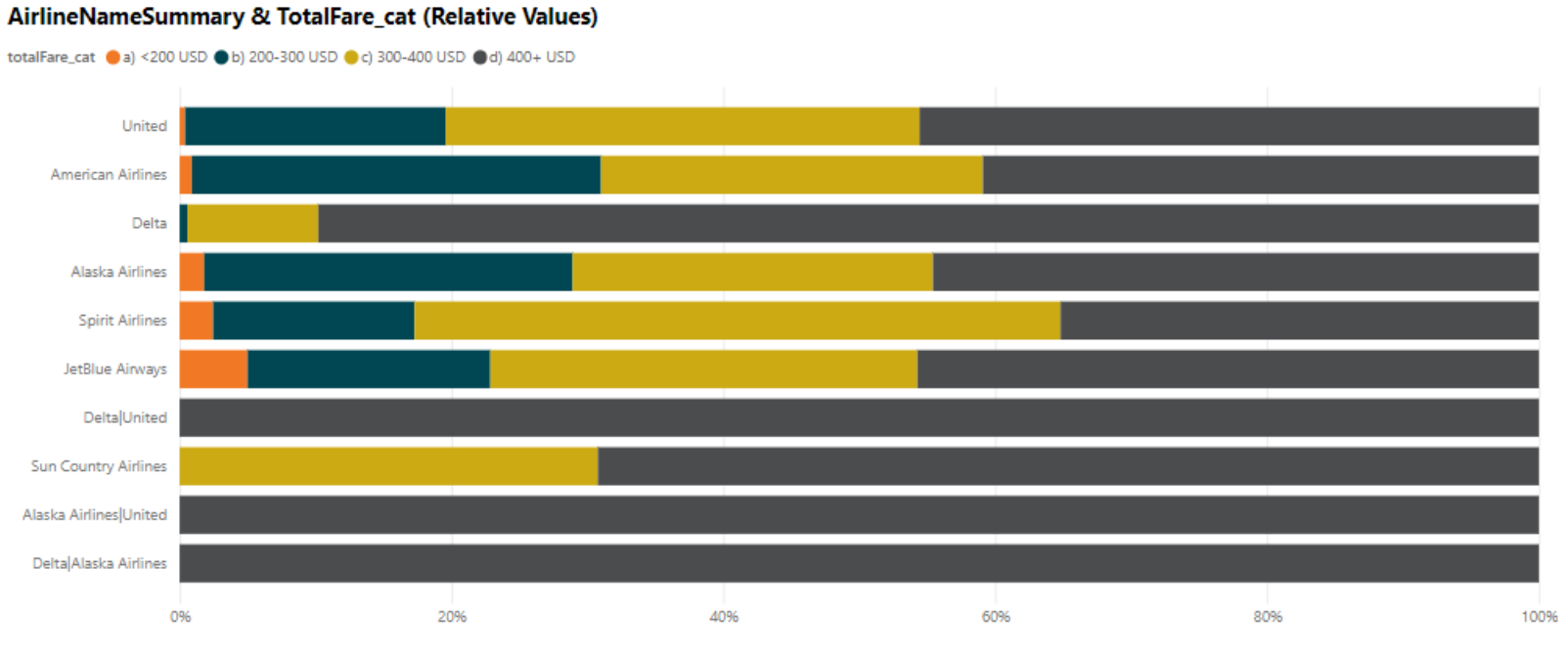
```
CleverMiner task processing summary:

Task type : SD4ftMiner
Number of verifications : 6264
Number of rules : 5
Total time needed : 00h 00m 08s
Time of data preparation : 00h 00m 02s
Time of rule mining : 00h 00m 06s


List of rules:
RULEID BASE1 BASE2 RatioConf DeltaConf Rule
     1   134    72    1.533    +0.228  destinationAirport(DEN) & startingAirport(SFO) => totalFare_cat(d) 400+ USD) | --- : AirlineNameSummary(American Airlines) x AirlineNameSummary(Alaska Airlines)
     2   317    74    2.014    +0.452  destinationAirport(EWR) & startingAirport(LAX) => totalFare_cat(d) 400+ USD) | --- : AirlineNameSummary(Delta) x AirlineNameSummary(Alaska Airlines)
     3   140    62    1.629    +0.386  destinationAirport(OAK) & startingAirport(ORD) => totalFare_cat(d) 400+ USD) | --- : AirlineNameSummary(United|Alaska Airlines) x AirlineNameSummary(Alaska Airlines)
     4   327    51    1.627    +0.386  destinationAirport(DFW) & startingAirport(OAK) => totalFare_cat(d) 400+ USD) | --- : AirlineNameSummary(United|Delta) x AirlineNameSummary(Alaska Airlines)
     5   143    62    1.629    +0.386  destinationAirport(OAK) & startingAirport(ORD) => totalFare_cat(d) 400+ USD) | --- : AirlineNameSummary(United|Delta) x AirlineNameSummary(Alaska Airlines)



Rule id : 1

Base1 :   134 Base2 :    72  Relative base 1 : 0.001 Relative base 2 : 0.000 CONF1 : 0.657  CONF2 : +0.429  Delta Conf : +0.228 Ratio Conf : +1.533

Cedents:
  antecedent : destinationAirport(DEN) & startingAirport(SFO)
  succcedent : totalFare_cat(d) 400+ USD)
  condition  : ---
  first set  : AirlineNameSummary(American Airlines)
  second set : AirlineNameSummary(Alaska Airlines)

Fourfold tables:
FRST|  S  |  ¬S |  SCND|  S  |  ¬S |
----|-----|-----|  ----|-----|-----|
 A  |  134|   70|   A  |   72|   96|
----|-----|-----|  ----|-----|-----|
¬A  |12285|64308|  ¬A  | 3859| 3280|
----|-----|-----|  ----|-----|-----|
```

- $\textbf{RatioConf} = \frac{134/(134+70)}{72/(72+96)} = \frac{0{,}657}{0{,}429} = 1{,}533$
- Relativní četnost letenek ze SFO do DEN, které stojí více než 400 USD, je u American Airlines 1,533x vyšší než u Alaska Airlines.

CENY LETENEK & AEROLINKY – INTERPRETACE I

- Především se to týká letenek z Los Angeles do Newarku nebo letenek ze San Francisca do Denveru.
- Například relativní četnost drahých letenek mezi těmi z Los Angeles do Newarku je v případě aerolinky Delta přibližně 2x vyšší než relativní četnost drahých letenek na této trase se společností Alaska Airlines.

Airlines and Flight Prices



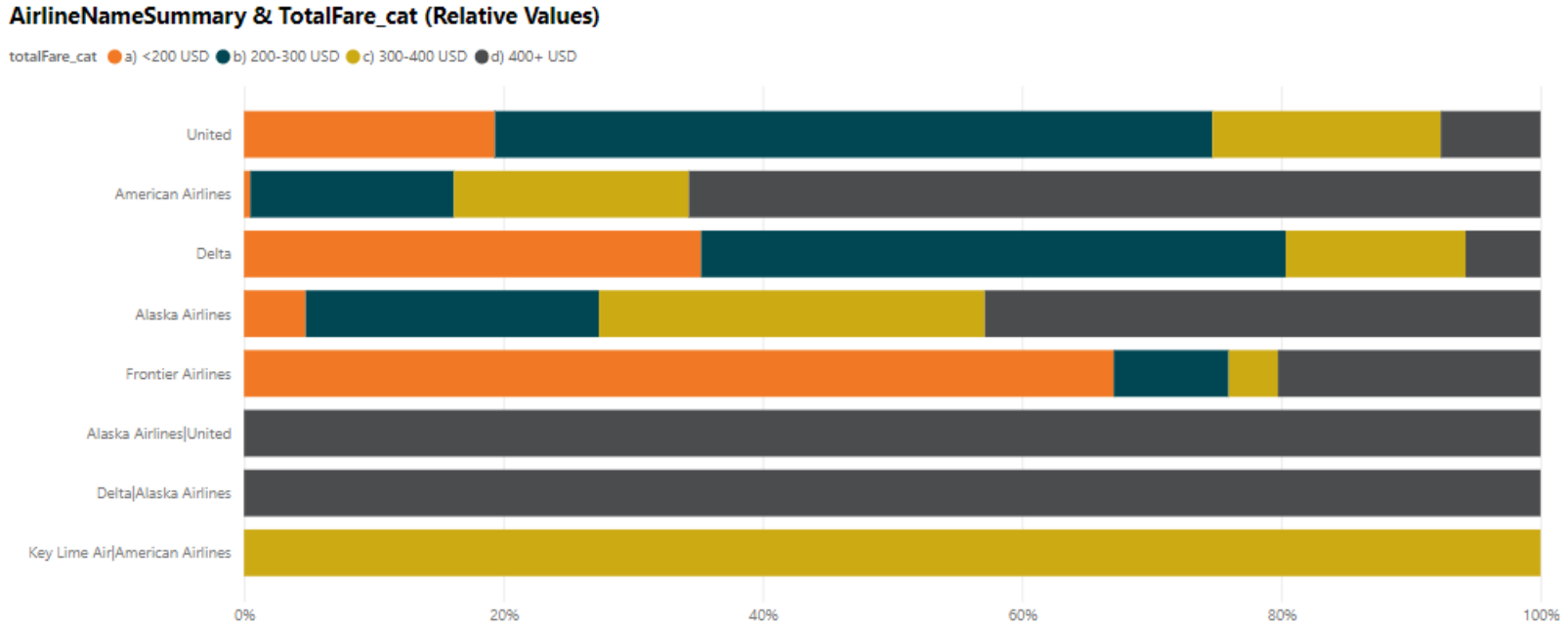
startingAirport

ATL	CLT	DFW	IAD	MIA	ORD	SFO
BOS	DEN	DTW	LAX	OAK	PHL	

destinationAirport

ATL
BOS
CLT
DEN
DFW
DTW
EWR
IAD
JFK
LGA
MIA
OAK
ORD
PHL
SFO

Airlines and Flight Prices



startingAirport

ATL	CLT	DTW	IAD	LAX	MIA	ORD	SFO
BOS	DFW	EWR	JFK	LGA	OAK	PHL	

destinationAirport

ATL
BOS
CLT
DEN
DFW
DTW
EWR
IAD
JFK
LAX
LGA
MIA
OAK
ORD
PHL

SD4FT-MINER – ZÁVĚR

- Ceny letenek se mohou na jednotlivých trasách napříč leteckými společnostmi velmi lišit. Bylo by vhodné detailněji prozkoumat, jaká cena by byla na dané trase ideální pro optimalizaci (maximalizaci) zisku při co nejefektivnější eliminaci konkurence.

04

PREDIKTIVNÍ MODELY

RANDOM FOREST

- Random Forest metoda využívá ke klasifikaci metodu Rozhodovacího stromu.
- Pomocí této metody bude vytvořen predikční model za účelem predikce cenové kategorie letenky na základě vybraných vysvětlujících atributů.
- Kromě klasického Random Forest predikčního modelu bude následně ještě využito nadstavby v Pythonu, která slouží k vylepšení výsledků modelu díky tzv. „ladění“ vstupních parametrů.

VYSVĚTLUJÍCÍ ATRIBUTY

- startingAirport, destinationAirport
- Weekday, dep_time_cat, arr_time_cat, elapsedDays
- DaysToFlight_cat, seatsRemaining
- isBasicEconomy, CabinCodeSummary
- Tot_Trav_Duration_cat, totalTravelDistance_cat
- Wait_time_cat, NumberOfTransfers
- AirlineNameSummary, AirlineNameCount
- tavg_cat, prcp_cat, wspd_cat

PŘÍPRAVA DAT

- Kategoriální vysvětlující atributy ze vstupní datové matice byly nejprve transformovány do podoby dummy proměnných.
- Dále proběhlo rozdělení na matici vysvětlujících proměnných X a vektor vysvětlované proměnné ‚totalFare_cat' jako y.
- Data byla následně rozdělena na trénovací a testovací sadu v poměru 8:2.
- Rozložení hodnot vysvětlovaného atributu totalFare_cat mezi jednotlivými kategoriemi je rovnoměrné, nebylo tudíž nutné volit žádnou over/undersamplingovou metodu.

```
encoded_data = pd.get_dummies( matice.drop(['totalFare_cat','a_arr_index','index', 'flightDate', 'increaseFare_cat', 'EquipmentDescriptionSummary'], axis=1) , drop_first=True)
encoded_data.head()
```

	elapsedDays	isBasicEconomy	isNonStop	seatsRemaining	NumberOfTransfers	AirlineNameCount	startingAirport_BOS	startingAirport_CLT	startingAirport_DEN	startingAirport_DFW	...	prcp_cat_e) 30+
0	0.0	False	True	9.0	0.0	1	False	False	False	False	...	False
1	0.0	False	True	4.0	0.0	1	False	False	False	False	...	False
2	0.0	False	True	9.0	0.0	1	False	False	False	False	...	False
3	0.0	False	True	8.0	0.0	1	False	False	False	False	...	False
4	0.0	False	True	9.0	0.0	1	False	False	False	False	...	False

5 rows × 102 columns

```
X = encoded_data  # Features
y = matice['totalFare_cat']  # Target variable
```

```
# Rozdělení datasetu na trénovací a testovací v poměru 80:20. Random state, který zajistí reproducibilitu je 42. Parametr stratify=y zajistí,
# že budou proporce datasetu vyváženy
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)
```

- Parametry RandomForestClassifieru byly nastaveny následovně:
  - n_estimators jako počet stromů v lese roven 50
  - Random_state pro zajištění reproducibility roven 42
  - Ostatní parametry ponechány na defaultních hodnotách

```
rf_classifier = RandomForestClassifier(n_estimators=50, random_state=42)

rf_classifier.fit(X_train, y_train)

y_pred_rf = rf_classifier.predict(X_test)

accuracy_rf = accuracy_score(y_test, y_pred_rf)
print(f"Accuracy: {accuracy_rf}")
```

```
Accuracy: 0.85394
```

```
report_rf = classification_report(y_test, y_pred_rf)
print(report_rf)
```

```
                precision    recall  f1-score   support

   a) >200 USD       0.89      0.90      0.90     12233
b) 200-300 USD       0.81      0.81      0.81     12015
c) 300-400 USD       0.79      0.77      0.78     10689
   d) 400+ USD       0.90      0.91      0.91     15063

      accuracy                           0.85     50000
     macro avg       0.85      0.85      0.85     50000
  weighted avg       0.85      0.85      0.85     50000
```

- Výsledná **Accuracy** modelu vyšla **85,39 %**, což je vcelku dobrý výsledek.
- Z povahy modelu vyplývá, že hlavní vyhodnocovací metrikou bude v našem případě především právě Accuracy predikčního modelu, případně i hodnota **AUC** nebo **F1-score** kombinující ve svém výpočtu Precision a Recall.
- Obzvlášť úspěšnost predikce u okrajových cenových kategorií je velmi úspěšná.

```
rf_classifier = RandomForestClassifier(n_estimators=50, random_state=42)

rf_classifier.fit(X_train, y_train)

y_pred_rf = rf_classifier.predict(X_test)

accuracy_rf = accuracy_score(y_test, y_pred_rf)
print(f"Accuracy: {accuracy_rf}")
```

```
Accuracy: 0.85394
```

```
report_rf = classification_report(y_test, y_pred_rf)
print(report_rf)
```

```
                precision    recall  f1-score   support

   a) >200 USD       0.89      0.90      0.90     12233
b) 200-300 USD       0.81      0.81      0.81     12015
c) 300-400 USD       0.79      0.77      0.78     10689
   d) 400+ USD       0.90      0.91      0.91     15063

      accuracy                           0.85     50000
     macro avg       0.85      0.85      0.85     50000
  weighted avg       0.85      0.85      0.85     50000
```

INTERPRETACE

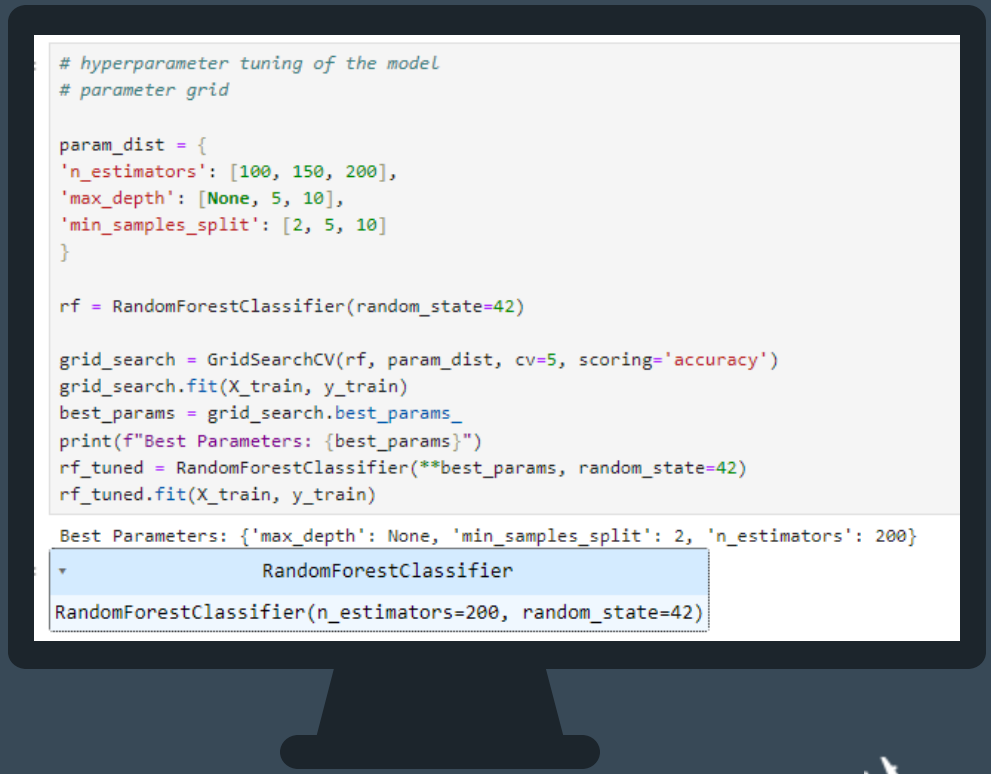
- Největší vliv na predikci cenové kategorie letenky mají v tomto modelu atributy seatsRemaining, AirlineNameCount, isBasicEconomy nebo totalTravelDistance_cat pro hodnotu c) 2000+

```python
# Model interpretation
importances = rf_classifier.feature_importances_

feature_importance = list(zip(importances, X.columns))  # Replace 'X' with your feature DataFrame
feature_importance.sort(reverse=True)

for importance, feature in feature_importance:
    print(f"Feature: {feature}, Importance: {importance}")
```

```
Feature: seatsRemaining, Importance: 0.07958309361135785
Feature: AirlineNameCount, Importance: 0.04322402748835587
Feature: isBasicEconomy, Importance: 0.03341949768887753
Feature: totalTravelDistance_cat_c) more than 2000 miles, Importance: 0.03209539528656629
Feature: DaysToFlight_cat_d) 11+ Days, Importance: 0.023309635505606902
Feature: arr_time_cat_b) Afternoon, Importance: 0.021890388556846857
Feature: dep_time_cat_b) Afternoon, Importance: 0.021843215820628623
Feature: DaysToFlight_cat_c) 8 - 10 Days, Importance: 0.020762267791544997
Feature: NumberOfTransfers, Importance: 0.019395214164214806
Feature: AirlineNameSummary_American Airlines, Importance: 0.018391078907839874
Feature: arr_time_cat_c) Evening, Importance: 0.018000422959156832
Feature: Weekday_2 Tue, Importance: 0.01709503024373418
Feature: startingAirport_LGA, Importance: 0.016848849519332323
Feature: dep_time_cat_d) Night, Importance: 0.016793369051041576
Feature: Weekday_7 Sun, Importance: 0.016513226839977326
```

HYPERPARAMETER TUNING I

- Následně byly vyzkoušeny i jiné Random Forest modely pomocí funkce GridSearchCV pro ladění modelů.
- Vzhledem k velkému počtu vysvětlujících atributů byl tento tuning časově velmi náročný.



```
# hyperparameter tuning of the model
# parameter grid

param_dist = {
'n_estimators': [100, 150, 200],
'max_depth': [None, 5, 10],
'min_samples_split': [2, 5, 10]
}

rf = RandomForestClassifier(random_state=42)

grid_search = GridSearchCV(rf, param_dist, cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)
best_params = grid_search.best_params_
print(f"Best Parameters: {best_params}")
rf_tuned = RandomForestClassifier(**best_params, random_state=42)
rf_tuned.fit(X_train, y_train)
```

```
Best Parameters: {'max_depth': None, 'min_samples_split': 2, 'n_estimators': 200}
```

```
RandomForestClassifier
RandomForestClassifier(n_estimators=200, random_state=42)
```

HYPERPARAMETER TUNING II

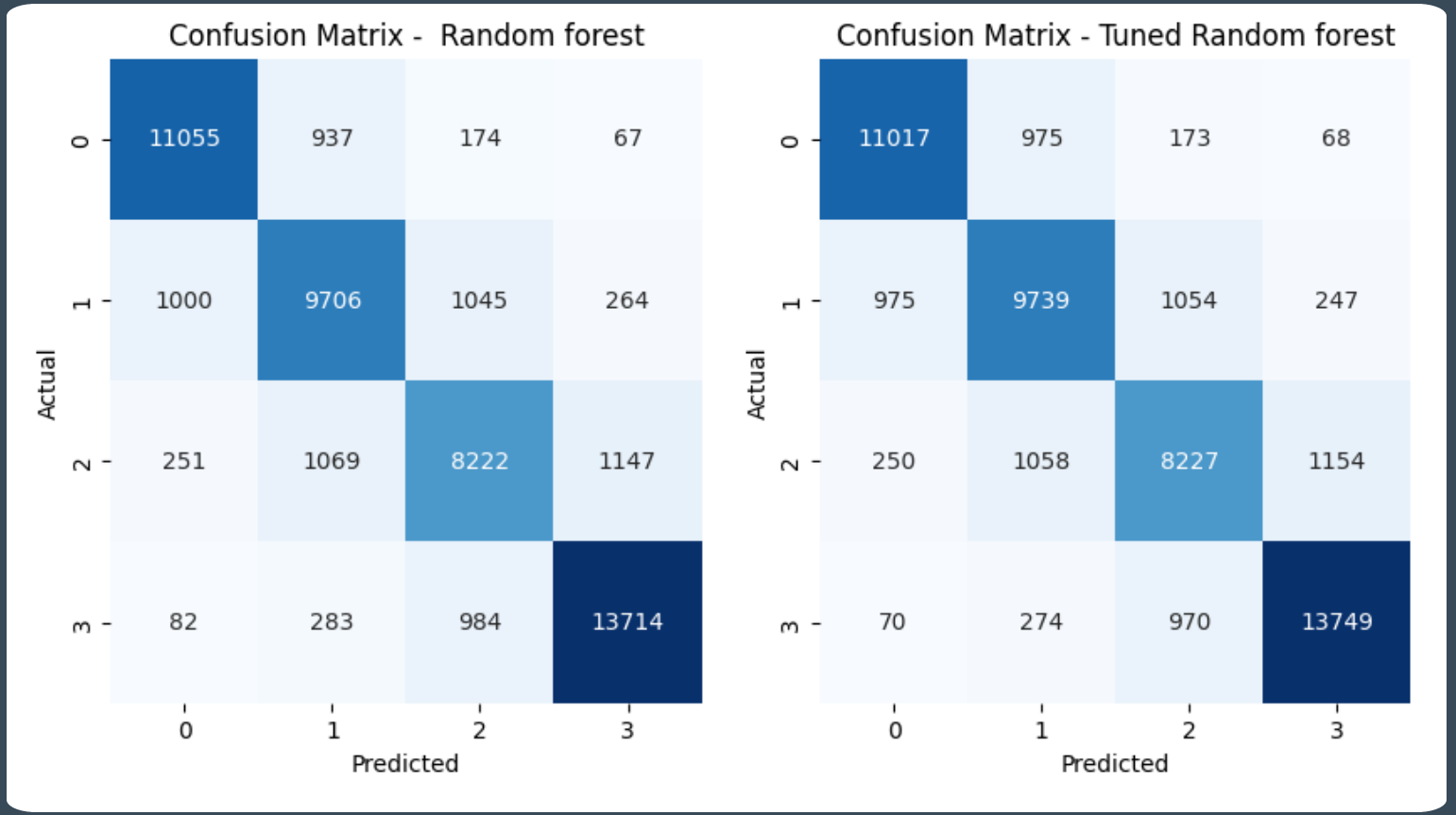
- Navzdory vysoké časové náročnosti funkce nenašla model, který by byl výrazně lepší oproti předchozímu modelu.
- Accuracy: 85.46 % (což není zlepšení ani o 0.1 procentního bodu)

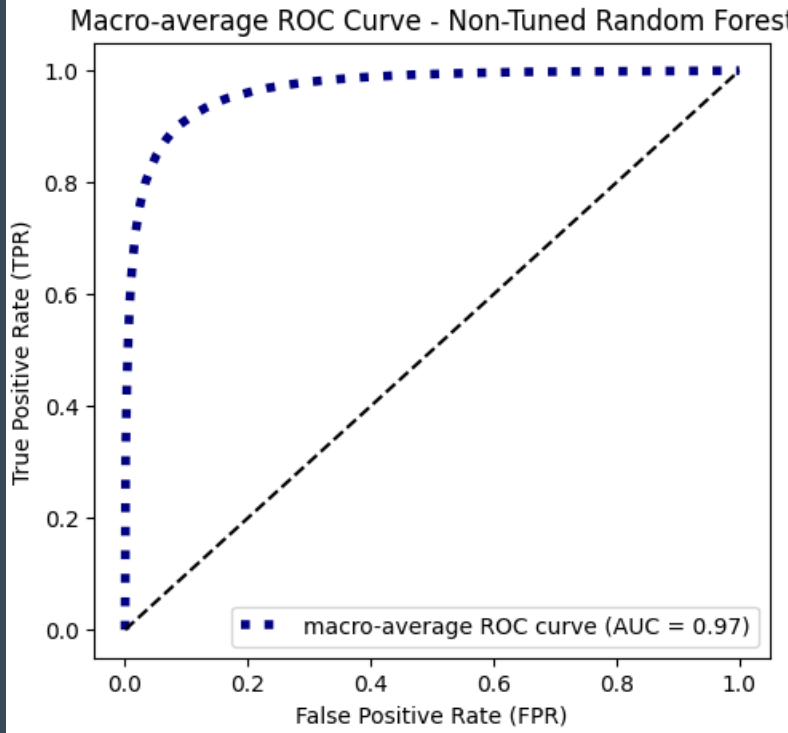
```
y_pred_rf_tuned = rf_tuned.predict(X_test)

accuracy_rf_tuned = accuracy_score(y_test, y_pred_rf_tuned)
print(f"Accuracy: {accuracy_rf_tuned}")
```

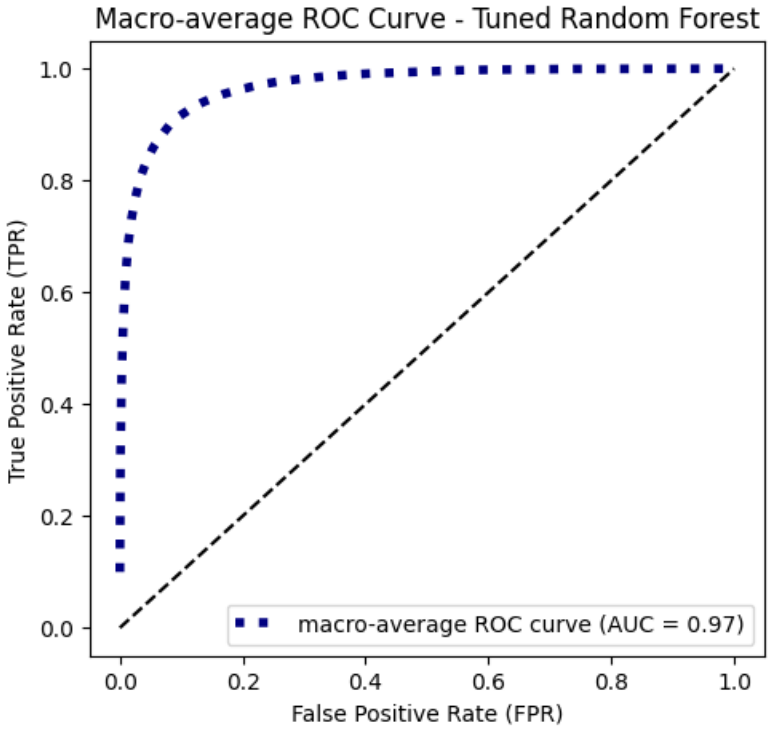
```
Accuracy: 0.85464
```

```
report_rf_tuned = classification_report(y_test, y_pred_rf_tuned)
print(report_rf_tuned)
```

```
                precision    recall  f1-score   support

   a) >200 USD       0.89      0.90      0.90     12233
b) 200-300 USD       0.81      0.81      0.81     12015
c) 300-400 USD       0.79      0.77      0.78     10689
   d) 400+ USD       0.90      0.91      0.91     15063

      accuracy                           0.85     50000
     macro avg       0.85      0.85      0.85     50000
  weighted avg       0.85      0.85      0.85     50000
```


Confusion Matrix - Random forest
Actual
Predicted
0
1
2
3
11055
937
174
67
1000
9706
1045
264
251
1069
8222
1147
82
283
984
13714
Confusion Matrix - Tuned Random forest
Actual
Predicted
0
1
2
3
11017
975
173
68
975
9739
1054
247
250
1058
8227
1154
70
274
970
13749


Macro-average ROC Curve - Non-Tuned Random Forest
True Positive Rate (TPR)
False Positive Rate (FPR)
macro-average ROC curve (AUC = 0.97)
0.0
0.2
0.4
0.6
0.8
1.0


Macro-average ROC Curve - Tuned Random Forest
True Positive Rate (TPR)
False Positive Rate (FPR)
macro-average ROC curve (AUC = 0.97)
0.0
0.2
0.4
0.6
0.8
1.0

VYHODNOCENÍ PREDIKTIVNÍCH MODELŮ

RANDOM FOREST

Accuracy: 85.39 %

AUC: 0.97

Nízká časová náročnost

RF TUNED

Accuracy: 85.46 %

AUC: 0.97

Vysoká časová náročnost

VYHODNOCENÍ PREDIKTIVNÍCH MODELŮ – ZÁVĚR

- Hodnoty vyhodnocovacích metrik velmi podobné u obou modelů.
- Z důvodu zachování efektivnosti analýzy bude jako lepší predikční model vyhodnocen původní nevyladěný Random Forest model.

ZÁVĚR

Shrnutí nově nabytých znalostí z předchozích analýz

SHRNUTÍ

- Cílem této práce byla analýza letenek pro vybrané destinace od různých leteckých společností pro různé hodnoty dalších atributů.
- Data byla zkoumána především z finanční stránky, většina analytických otázek se proto týkala především cen letenek, tzn. byly zkoumány především vztahy atributu totalFare_cat s ostatními dostupnými atributy.
- Účelem této práce bylo pomoci s průzkumem pro začínající leteckou společnost, která zvažuje, jak vysoké ceny letenek by měla stanovit či na jaké trasy by bylo vhodné se soustředit.

ZÁVĚR – CENY LETENEK

- Nejprve byly hledány případy, při kterých byly ceny letenek nečekaně vyšší nebo nižší oproti běžné situaci.
- Zjevnější anomálie v cenách letenek a celkově základní souvistosti v datech byly objeveny již v rámci prvotního průzkumu struktury dat pomocí BI analýzy.
- V Data Science části projektu bylo cílem najít případně některé další trendy či odlišnosti, které se nepodařilo odhalit v BI analýze, ale také především najít takové situace, za kterých pravidla a trendy nalezené pomocí Power BI byly porušeny.

CENY LETENEK – S PROUDEM NEBO PROTI?

- Tato analýza posloužila zmiňované nastupující letecké společnosti mimo jiné jako průzkum trhu.
- Byly nalezeny některé typické vzorce v chování konkurenčních společností při oceňování jednotlivých letů.
- Aerolinka má nyní 2 možnosti ohledně nastavování cen svých letenek.

„S PROUDEM“

- Letenky, které jsou u konkurenčních aerolinek za daných podmínek drahé, nastavím také na vyšší cenu, jelikož si to mohu dovolit, zjevně tyto ceny zákaznící akceptují i od ostatních společností.

„PROTI PROUDU“

- Letenky, které jsou u konkurenčních aerolinek za daných podmínek drahé, prodám naopak za nižší ceny (pokud to náklady dovolí) a získám tak potenciální konkurenční výhodu oproti ostatním společnostem.
- Naopak levné letenky mohu zkusit prodávat draze, pokud samozřejmě bude jejich prodej i tak dostatečně velký a zisky z nich tak pokryjí všechny náklady.

ZÁVĚR II

- Volba jedné z variant může být individuální, tedy odlišná pro různé trasy.
- Volba závisí především na podrobnější analýze nákladů společnosti. Cílem tohoto projektu bylo pouze dát společnosti návrhy k oblastem, které by z tohoto pohledu bylo možné podrobněji prozkoumat.

ZÁVĚR III – POTENCIÁL V PŘETÍŽENÝCH TRASÁCH

- Dalším velmi podstatným zjištěním v rámci tohoto projektu bylo objevení potenciálně přetížených tras. Právě na nich by tato nová aerolinka mohla najít své uplatnění posílením těchto tras. Začínající letecké společnosti by tato informace mohla velmi pomoci především v nelehkých začátcích fungování na trhu.

ZÁVĚR IV

- Prediktivní model by mohl posloužit například jako prvotní "nástřel" nových cen dle odhadu získaného na základě zadaných vstupních parametrů

- Na samotný závěr ještě poznamenejme, že potenciál dat ještě nebyl zcela vyčerpán. Bylo by možné dále využít některé doposud nevyužité/málo využité atributy k nalezení dalších pravidel, výjimek nebo souvislostí.

KONEC

RESOURCES I

- https://air.flyingway.com/books/xls/airport-codes.xls
- https://www.kaggle.com/datasets/justinmitchel/flightprices-min?resource=download&select=itineraries-min-250k.csv
- https://www.transtats.bts.gov/DL_SelectFields.aspx?gnoyr_VQ=FLL&QO_fu146_anzr=
- https://dev.meteostat.net/python/
- https://www.expedia.com/Flights
- https://www.budgetair.com/en_ca/blog/what-are-the-different-cabin-classes-on-airplanes
- https://simpleflying.com/how-airline-ticket-pricing-works/

RESOURCES II

- https://www.flightapi.io/blog/airline-pricing-strategies/
- https://www.w3.org/TR/NOTE-datetime
- https://www.chmi.cz/files/portal/docs/meteo/om/sivs/dest.html
- https://www.rmets.org/metmatters/beaufort-wind-scale

PHOTOS, ICONS AND TEMPLATE:

- https://slidesgo.com/theme/plane-flying-in-the-sky#search-Airplane&position-10&results-70
- Icon Pack | Aviation (flaticon.com)
- Free Photo | Place flying in sunset sky (freepik.com)
- Free Photo | Low angle tall chimney and airplane (freepik.com)
- Free Photo | Daytime skyscape (freepik.com)