hr-comma-sep

January 13, 2024

1. Die Hauptaufgabe besteht darin festzustellen wann und wieso Mitarbeiter das Unternehmen verlassen.
   - Eine ML-schreiben das durch das verhalten der Mitarbeiter erkennt wer als nächstes gehen könnte ’

Welche Biblitheken nutze ich, zur Bearbeitung meiner Analyse

[301]:
```
# Diese Biblotheken sind zur Bearbeitung und Visualisierung
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pylab as plt
# Hie später die Bibliotheken zur Erstellung der ML einfügen
```

1 1. ASK/Prepare

```
- Spalten betrachen
- was muss sich ändern
- Welche Spalten sind wichtig
- Einheitlichkeit überprüfen
- Mit welchen Datentypen arbeite ich
- Hat meine Datei identische Zeilen
- Index festlegen
```

[302]:
```
#Einlesung meiner Datei
df = pd.read_excel("HR_comma_sep Kopie.xlsm")
df
```

[302]:

	satisfaction_level	last_evaluation	number_project	\
0	0.38	0.53	2	
1	0.80	0.86	5	
2	0.11	0.88	7	
3	0.72	0.87	5	
4	0.37	0.52	2	
…	…	…	…	
14994	0.40	0.57	2	
14995	0.37	0.48	2	
14996	0.37	0.53	2	

```
14997                0.11             0.96                6
14998                0.37             0.52                2

       average_montly_hours  time_spend_company  Work_accident  left  \
0                       157                   3              0     1
1                       262                   6              0     1
2                       272                   4              0     1
3                       223                   5              0     1
4                       159                   3              0     1
…                       …                     …              …   …
14994                   151                   3              0     1
14995                   160                   3              0     1
14996                   143                   3              0     1
14997                   280                   4              0     1
14998                   158                   3              0     1

       promotion_last_5years Department  salary
0                          0      sales     low
1                          0      sales  medium
2                          0      sales  medium
3                          0      sales     low
4                          0      sales     low
…                        …          …      …
14994                      0    support     low
14995                      0    support     low
14996                      0    support     low
14997                      0    support     low
14998                      0    support     low

[14999 rows x 10 columns]
```

[303]:
```
df.columns
```

[303]:
```
Index(['satisfaction_level', 'last_evaluation', 'number_project',
       'average_montly_hours', 'time_spend_company', 'Work_accident', 'left',
       'promotion_last_5years', 'Department', 'salary'],
      dtype='object')
```

[304]:
```
#Spalten Namen sind nicht Einheitlich
df = df.rename(columns={"satisfaction_level": "Satisfaction_Level",
                        "last_evaluation": "Last_Evaluation",
                        "number_project": "Number_Project",
                        "average_montly_hours": "Average_Monthly_Hours",
                        "time_spend_company": "Time_Spend_Company",
                        "left": "Left",
                        "promotion_last_5years": "Promotion_Last_5Years",
                        "salary": "Salary"
```

```
                    })
```

[305]:
```
#Überprüfung der Daten (Objects = Strings)
df.dtypes
```

[305]:
```
Satisfaction_Level       float64
Last_Evaluation          float64
Number_Project             int64
Average_Monthly_Hours      int64
Time_Spend_Company         int64
Work_accident              int64
Left                       int64
Promotion_Last_5Years      int64
Department                object
Salary                    object
dtype: object
```

[306]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14999 entries, 0 to 14998
Data columns (total 10 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Satisfaction_Level     14999 non-null  float64
 1   Last_Evaluation        14999 non-null  float64
 2   Number_Project         14999 non-null  int64
 3   Average_Monthly_Hours  14999 non-null  int64
 4   Time_Spend_Company     14999 non-null  int64
 5   Work_accident          14999 non-null  int64
 6   Left                   14999 non-null  int64
 7   Promotion_Last_5Years  14999 non-null  int64
 8   Department             14999 non-null  object
 9   Salary                 14999 non-null  object
dtypes: float64(2), int64(6), object(2)
memory usage: 1.1+ MB
```

[307]:
```
df.sample(10)
```

[307]:

	Satisfaction_Level	Last_Evaluation	Number_Project	\
4631	0.92	0.79	5	
13720	0.75	0.87	4	
13026	0.83	0.73	4	
9846	0.67	0.50	5	
5578	0.47	0.55	2	
7564	0.95	0.90	2	

```
769                 0.42             0.46                2
6132                0.76             0.37                3
12791               0.82             0.49                4
6264                0.49             0.96                2

       Average_Monthly_Hours  Time_Spend_Company  Work_accident  Left  \
4631                     243                   3              1     0
13720                    146                   8              1     0
13026                    247                   2              0     0
9846                     219                   3              0     0
5578                     156                   2              0     0
7564                     129                   5              0     0
769                      141                   3              0     1
6132                     127                   4              0     0
12791                    276                   4              0     0
6264                     206                   2              0     0

       Promotion_Last_5Years  Department  Salary
4631                       0     support    high
13720                      0     support     low
13026                      0     support  medium
9846                       0          IT  medium
5578                       0  management  medium
7564                       0   marketing  medium
769                        0   marketing  medium
6132                       0       sales  medium
12791                      0     support     low
6264                       0   marketing     low
```

[308]:
```
df.shape
```

[308]:
```
(14999, 10)
```

[309]:
```
#Überprüfung der Mitarbeiter Pro Abteilung
df["Department"].value_counts()
```

[309]:
```
Department
sales          4140
technical      2720
support        2229
IT             1227
product_mng     902
marketing       858
RandD           787
accounting      767
hr              739
management      630
```

```
Name: count, dtype: int64
```

2. Prepare
   - Die oben genannten Punkte durchsetzen
   - Alles einheitlich und dynamisch gestallten

[310]:
```
#duplicat(keep=True) => geht df durch merkt sich die  Zeilen falls eine zeile␣
↪ein Duplicat ist gibt er diese aus

dobbelvalues= df[df.duplicated()]
dobbelvalues
```

[310]:

	Satisfaction_Level	Last_Evaluation	Number_Project	\
396	0.46	0.57	2	
866	0.41	0.46	2	
1317	0.37	0.51	2	
1368	0.41	0.52	2	
1461	0.42	0.53	2	
…	…	…	…	
14994	0.40	0.57	2	
14995	0.37	0.48	2	
14996	0.37	0.53	2	
14997	0.11	0.96	6	
14998	0.37	0.52	2	

	Average_Monthly_Hours	Time_Spend_Company	Work_accident	Left	\
396	139	3	0	1	
866	128	3	0	1	
1317	127	3	0	1	
1368	132	3	0	1	
1461	142	3	0	1	
…	…	…	…	…	
14994	151	3	0	1	
14995	160	3	0	1	
14996	143	3	0	1	
14997	280	4	0	1	
14998	158	3	0	1	

	Promotion_Last_5Years	Department	Salary
396	0	sales	low
866	0	accounting	low
1317	0	sales	medium
1368	0	RandD	low
1461	0	sales	low
…	…	…	…
14994	0	support	low
14995	0	support	low

```
14996                      0     support     low
14997                      0     support     low
14998                      0     support     low

[3008 rows x 10 columns]
```

[311]:
```
#Oben wurden die Doppeltenwerte ermittelt, wir haben sie unserer Dataframe␣
 ↪entzogen
df = df.drop_duplicates()
df
```

[311]:
```
       Satisfaction_Level  Last_Evaluation  Number_Project  \
0                    0.38             0.53               2
1                    0.80             0.86               5
2                    0.11             0.88               7
3                    0.72             0.87               5
4                    0.37             0.52               2
…                     …                …               …
11995                0.90             0.55               3
11996                0.74             0.95               5
11997                0.85             0.54               3
11998                0.33             0.65               3
11999                0.50             0.73               4

       Average_Monthly_Hours  Time_Spend_Company  Work_accident  Left  \
0                        157                   3              0     1
1                        262                   6              0     1
2                        272                   4              0     1
3                        223                   5              0     1
4                        159                   3              0     1
…                        …                     …              …   …
11995                    259                  10              1     0
11996                    266                  10              0     0
11997                    185                  10              0     0
11998                    172                  10              0     0
11999                    180                   3              0     0

       Promotion_Last_5Years  Department  Salary
0                          0       sales     low
1                          0       sales  medium
2                          0       sales  medium
3                          0       sales     low
4                          0       sales     low
…                          …          …     …
11995                      1  management    high
11996                      1  management    high
11997                      1  management    high
```

```
11998                      1   marketing    high
11999                      0          IT     low

[11991 rows x 10 columns]
```

[400]:
```
df.sample(60)
```

[400]:

	Satisfaction_Level	Last_Evaluation	Number_Project	\
1643	0.09	0.83	6	
3301	0.49	0.69	2	
755	0.37	0.48	2	
5491	0.78	0.90	4	
8997	0.50	0.72	3	
9397	0.49	0.83	3	
9892	0.50	0.91	4	
10585	0.79	0.77	3	
2746	0.94	0.90	2	
5375	0.67	0.97	4	
2536	0.61	0.59	5	
11016	0.62	0.79	4	
4434	0.66	0.68	4	
4655	0.80	0.86	3	
3808	0.85	0.92	4	
4237	0.62	0.70	5	
2171	0.64	0.60	3	
8419	0.55	0.97	5	
4473	0.28	0.83	5	
8749	0.15	0.84	3	
7046	0.43	0.86	5	
2074	0.45	0.66	3	
11238	0.39	0.89	3	
1421	0.86	0.93	5	
7949	0.76	0.99	3	
1630	0.10	0.86	6	
1392	0.39	0.57	2	
229	0.78	0.98	5	
3256	0.95	0.50	4	
8494	0.52	0.83	3	
5441	0.50	0.60	5	
821	0.74	0.93	5	
4312	0.59	0.90	2	
8754	0.66	0.69	3	
4515	0.53	0.73	4	
2774	0.88	0.63	3	
8011	0.66	0.93	4	
2750	0.91	0.67	2	
6765	0.96	1.00	5	

7687	0.26	0.70	3
10727	0.97	0.93	3
1364	0.41	0.52	2
6780	0.83	0.71	3
9485	0.42	0.86	3
7929	0.49	0.87	3
1157	0.42	0.52	2
6935	0.91	0.55	3
8007	0.75	0.74	3
6004	0.82	0.88	4
8788	0.67	0.83	3
4836	0.50	0.84	3
6083	0.19	0.76	3
2843	0.50	0.95	5
10989	0.17	0.55	6
7433	0.20	0.80	6
5494	0.93	0.49	5
3380	0.63	0.93	4
3689	0.62	0.63	2
6889	0.71	0.93	3
10716	0.98	0.91	3

	Average_Monthly_Hours	Time_Spend_Company	Work_accident	Left	\
1643	295	5	0	1	
3301	147	2	0	0	
755	160	3	0	1	
5491	104	4	0	0	
8997	182	2	1	0	
9397	172	2	0	0	
9892	148	2	0	0	
10585	201	6	1	0	
2746	263	3	0	0	
5375	186	3	0	0	
2536	271	2	0	0	
11016	268	6	0	0	
4434	152	3	0	0	
4655	136	2	0	0	
3808	151	3	1	0	
4237	270	3	0	0	
2171	143	3	0	0	
8419	125	4	0	0	
4473	279	4	0	0	
8749	201	6	0	0	
7046	125	3	1	0	
2074	111	4	0	0	
11238	188	5	0	0	
1421	241	5	1	1	

7949	166	3	0	0
1630	288	4	0	1
1392	157	3	0	1
229	239	6	0	1
3256	242	2	0	0
8494	153	2	0	0
5441	216	3	0	0
821	244	5	0	1
4312	219	2	1	0
8754	257	2	0	0
4515	147	3	0	0
2774	257	3	0	0
8011	187	2	0	0
2750	255	4	0	0
6765	152	4	0	0
7687	238	6	0	0
10727	153	2	0	0
1364	147	3	0	1
6780	243	2	1	0
9485	160	4	1	0
7929	212	2	0	0
1157	141	3	1	1
6935	223	3	0	0
8007	186	3	1	0
6004	259	3	0	0
8788	220	3	0	0
4836	156	4	0	0
6083	107	5	0	0
2843	137	3	0	0
10989	240	6	0	0
7433	251	5	0	0
5494	167	3	0	0
3380	238	4	0	0
3689	123	2	0	0
6889	287	5	0	0
10716	165	2	1	0

	Promotion_Last_5Years	Department	Salary
1643	0	technical	low
3301	0	sales	medium
755	0	product_mng	low
5491	0	RandD	low
8997	0	product_mng	medium
9397	0	sales	medium
9892	0	technical	medium
10585	0	support	medium
2746	0	RandD	low

5375	0	hr	low
2536	0	sales	low
11016	0	sales	medium
4434	0	marketing	high
4655	0	RandD	medium
3808	0	IT	medium
4237	0	technical	high
2171	0	technical	medium
8419	0	sales	medium
4473	0	technical	medium
8749	0	support	medium
7046	0	sales	low
2074	0	sales	low
11238	0	support	low
1421	0	support	low
7949	0	sales	medium
1630	0	sales	medium
1392	0	sales	medium
229	0	marketing	low
3256	0	support	medium
8494	0	sales	medium
5441	0	sales	medium
821	0	technical	low
4312	0	technical	medium
8754	0	support	medium
4515	0	sales	low
2774	0	sales	low
8011	0	RandD	low
2750	0	accounting	low
6765	0	support	low
7687	0	support	medium
10727	0	technical	low
1364	0	product_mng	medium
6780	0	IT	low
9485	0	sales	low
7929	0	product_mng	medium
1157	0	sales	medium
6935	0	IT	low
8007	0	RandD	medium
6004	0	support	low
8788	0	sales	low
4836	0	accounting	low
6083	0	support	low
2843	0	sales	medium
10989	0	RandD	low
7433	0	accounting	medium
5494	0	RandD	medium

```
3380                     0        sales  medium
3689                     0        sales  medium
6889                     0        sales  medium
10716                    0           hr  medium
```

[312]:
```
#Überprüfung ob die df leere Zellen enthällt
df.isnull().sum()
```

[312]:
```
Satisfaction_Level       0
Last_Evaluation          0
Number_Project           0
Average_Monthly_Hours    0
Time_Spend_Company       0
Work_accident            0
Left                     0
Promotion_Last_5Years    0
Department               0
Salary                   0
dtype: int64
```

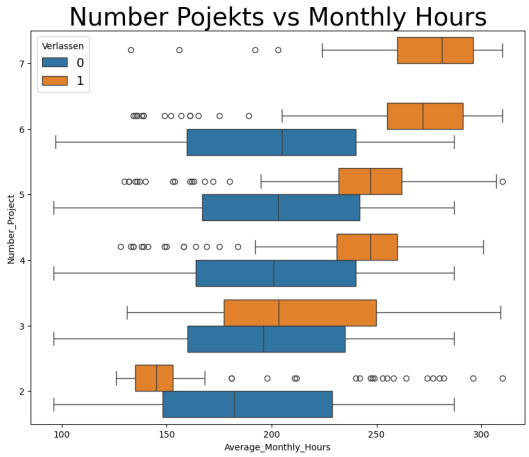
2 2. Analyse

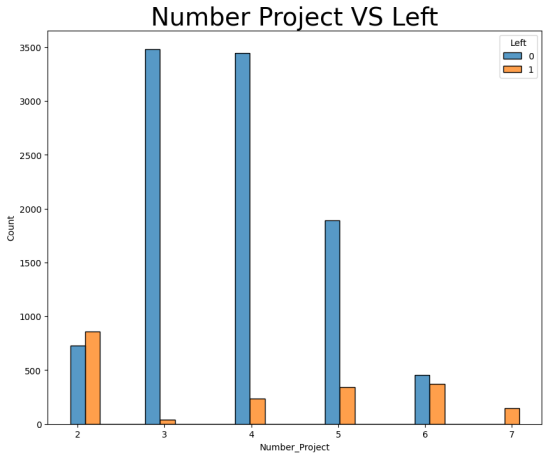
Left (Number_Project) - Vergleich zwichen anzahl Projekten und verlassen desn Unternehmens - Vergleich zwischen Number Projects und Average Monthly Hours

[313]:
```
# bezieht sich auf alle Mitarbeiter der Firma

#größe und axs bestimmen
fig ,ax = plt.subplots(1,2, figsize=(22,8))

#Vergleich erstllen für ax[0]
sns.boxplot(data=df,x="Average_Monthly_Hours",y="Number_Project",hue="Left",␣
 ↪ax=ax[0],orient="h",)
ax[0].invert_yaxis()     #Ändert die Reihnfolge der y-Achse
ax[0].set_title("Number Pojekts vs Monthly Hours", fontsize =28)
#ax[0].label_outer()

#neulabels = ["bleibt","geht"]
ax[0].legend(title="Verlassen", fontsize=14)


#Vergleich der number_project mit Angestellten die geblieben sind und die die␣
 ↪gegangen sind
stay = df[df["Left"] == 0]["Number_Project"]    #stay wird auf den Punkten von␣
 ↪["Number_Project"]  im Diagramm dargestellt
```

```
left = df[df["Left"] == 1]["Number_Project"]     ##left wird auf den Punkten von␣
 ↪["Number_Project"]  im Diagramm dargestellt

sns.histplot(data=df, x="Number_Project",hue="Left",ax=ax[1],multiple='dodge',␣
 ↪shrink=2)
ax[1].set_title("Number Project VS Left", fontsize=28)
```

[313]:
```
Text(0.5, 1.0, 'Number Project VS Left')
```





[314]:
```
# Beweis für meine Annahme das alle Mitarbeiter die mir 7 Projekten belastet␣
 ↪werden das Unternehmen verlassen

df[df["Number_Project"]==7]["Left"].value_counts()  #Nicht ganz Verstanden
```

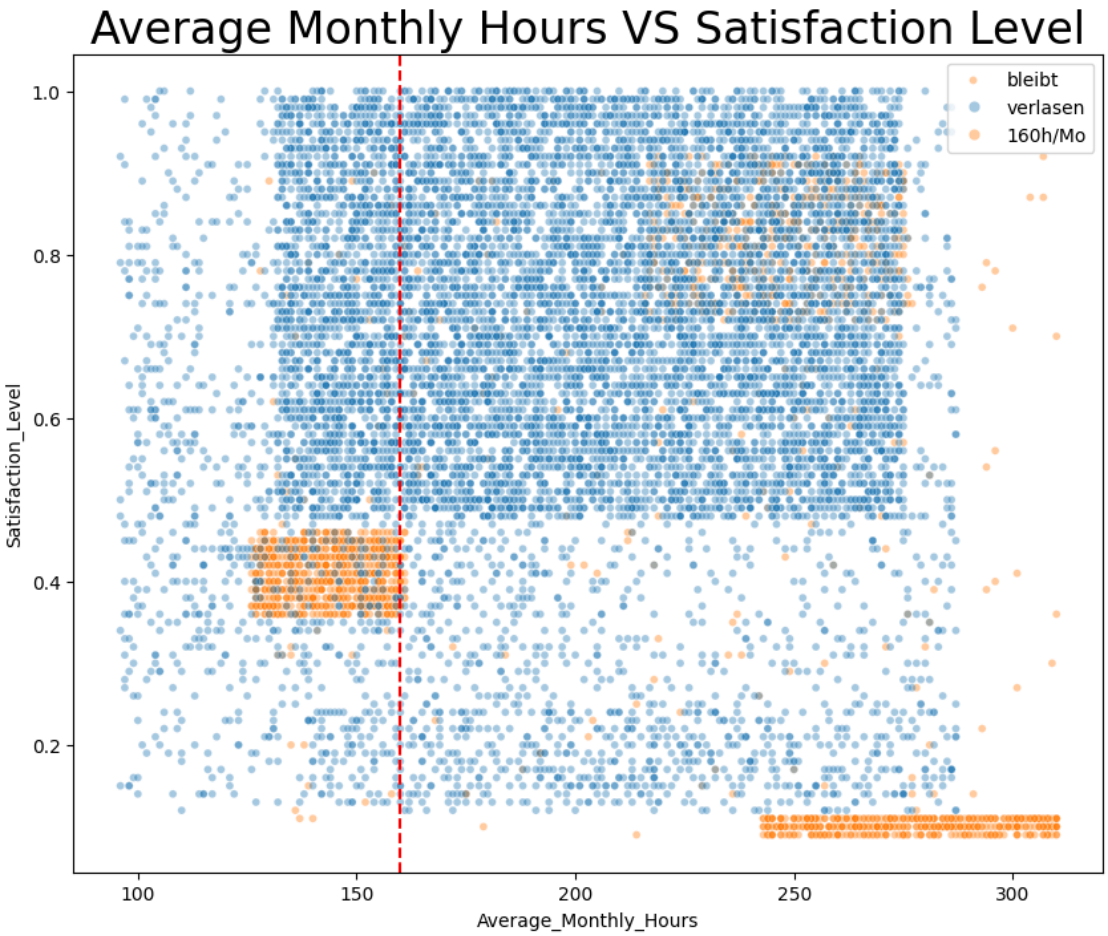
[314]:
```
Left
1    145
Name: count, dtype: int64
```

Hier erkennen wir ganz deutlich das Mitarbeiter die eine größere belastung ausgestzt sind auch das Unternehmen verlassen. - Alle die eine Überdurchschnittlich hohe Atbeitszeit aufweisen und zeitlich mit 6 oder mehr Projekten belastet sind - Epfehlenswert ist eine Belastung von 3- 4 Projekten

Averagemonthly hours versus the satisfaction levels

[315]:
```
plt.figure(figsize=(10,8))  # Die Größe meines Diagramms wird festgeleft
sns.
 ↪scatterplot(data=df,x="Average_Monthly_Hours",size=24,y="Satisfaction_Level",hue="Left",alp
 ↪4)  #Bedinungen/ Analyse
plt.title("Average Monthly Hours VS Satisfaction Level",fontsize=24)  #Title
plt.axvline(x=160,color="red", ls="--", label='160 h/mo')   #Einführung einer␣
 ↪Senlrechten Linie zur Kennzeichnung der der Stunden eines normalen Arbeiters
```

```
legend = plt.legend(labels=['bleibt', 'verlasen', "160h/Mo"])   #beschriftung␣
 ↪der Legende

#Legend noch bearbeiten
```



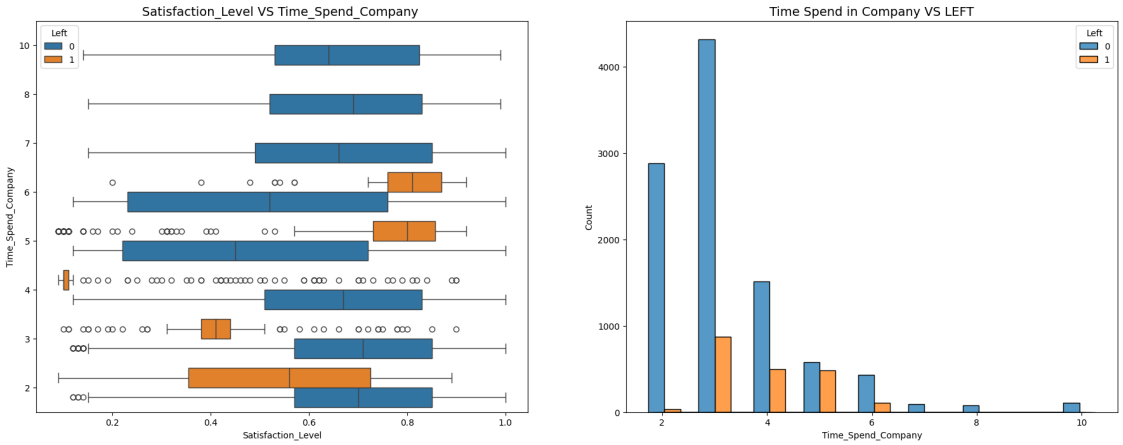
Ergebniss Es heben sich drei Felder besonders hervor 1. Arbeiter die sehr viel Arbeiten und undglücklich 2. Arbeiter die genau an der 160 Stunde an Grenzen, diese sind eventuell nicht genung ausgelastet 3. Arbeiter die eine hohe arbeitszeit aufweisen und eine hohe zufriedenheits grad

Satisfication_Level VS Time Spend in the company

[316]:
```
fig , ax=plt.subplots(1,2, figsize=(22,8))
sns.boxplot(data=df, x=df["Satisfaction_Level"],␣
 ↪y="Time_Spend_Company",hue="Left", ax=ax[0], orient="h")
ax[0].invert_yaxis()
ax[0].set_title("Satisfaction_Level VS Time_Spend_Company ",fontsize=14,)
```

```
# Create histogram showing distribution of `tenure`, comparing employees who␣
 ↪stayed versus those who left


sns.histplot(data=df, x="Time_Spend_Company",␣
 ↪hue=("Left"),multiple="dodge",shrink=7,ax=ax[1])
ax[1].set_title("Time Spend in Company VS LEFT",fontsize = 14)
```

[316]:
```
Text(0.5, 1.0, 'Time Spend in Company VS LEFT')
```



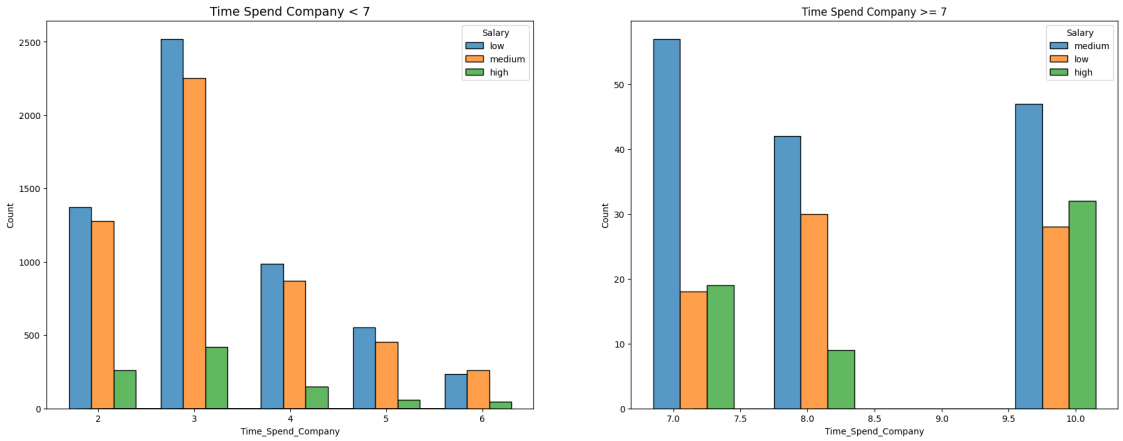
Das Ergebniss Mitarbeiter die das Unernehmen verlassen tuen das in zwischen dem 6 und 1 Jahr. Alle Mitarbeiter die das 7Jahr erreicht haben sind verlassen nicht mehr das UNternehmen.

[412]:
```
# Hier wird das Satisfaction_Level der der Personen die das Unternehmen die␣
 ↪verlassen(0,1 ) zusammengefast und durhc die agg Funktionen wieder ausgegeben
df.groupby("Left")["Satisfaction_Level"].agg(["mean","median"])
```

[359]:
```
#Next, you could examine salary levels for different tenures.
#Salery vs Time Spend Company
plt.subplots(1,2, figsize=(22,8))

# Timm_Spend_Company_Long/Short wurde als df festgehalten um es in einem plot␣
 ↪wieder geben zu können
Time_Spend_Company_Long = df[df["Time_Spend_Company"] > 6]
Time_Spend_Company_Short = df[df["Time_Spend_Company"] <= 6]

sns.histplot(data=Time_Spend_Company_Long,␣
 ↪x="Time_Spend_Company",hue="Salary",ax=ax[1],multiple="dodge",shrink=2)
ax[0].set_title("Time Spend Company < 7",fontsize=14)

sns.histplot(data=Time_Spend_Company_Short,␣
 ↪x="Time_Spend_Company",hue="Salary",ax=ax[0],multiple="dodge",shrink=8)
```

```
ax[1].set_title("Time Spend Company >= 7")
```

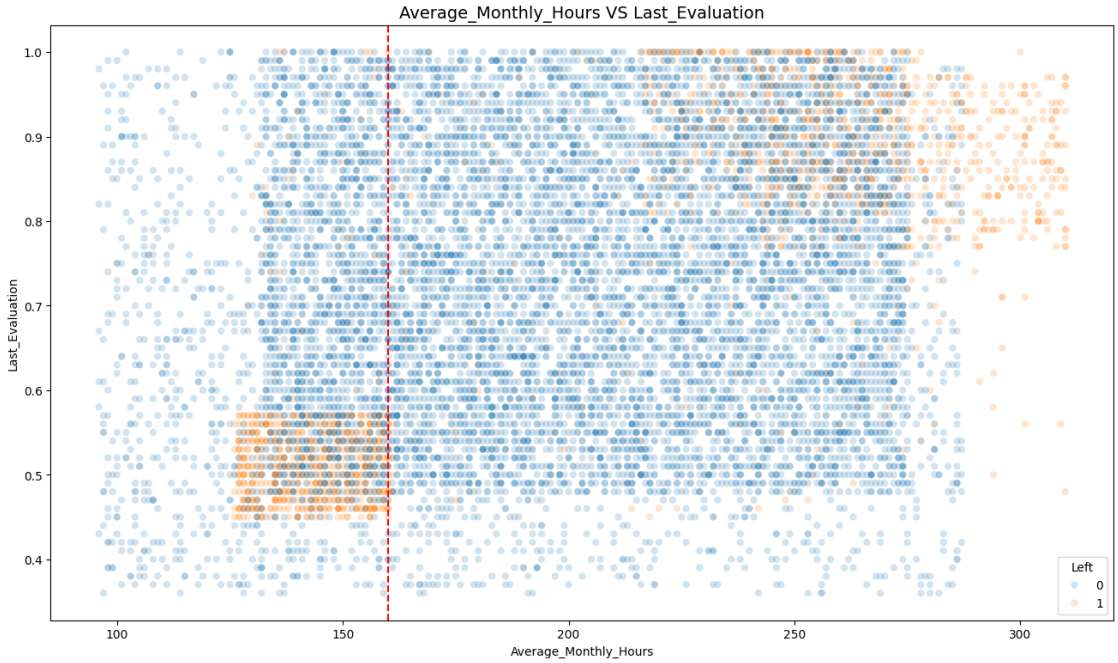
[359]:
```
Text(0.5, 1.0, 'Time Spend Company >= 7')
```



Ergebiss Wir erkennen das es keine überpropotionale Bezahlung für Mitarbeiter gibt die länger im Unternehmen arbeiten.

[389]:
```
"""Next, you could explore whether there's a correlation between working long␣
 ↪hours and receiving high
    evaluation scores. You could create a scatterplot of␣
 ↪`average_monthly_hours` versus `last_evaluation`."""

#average_monthly_hours` versus `last_evaluation
plt.figure(figsize=(16,9))
sns.scatterplot(data=df,x="Average_Monthly_Hours",␣
 ↪y="Last_Evaluation",hue="Left",alpha=0.2)
plt.title("Average_Monthly_Hours VS Last_Evaluation",fontsize=14)
plt.axvline(x = 160,color="red", ls="--")

#Legende muss nich bearbeitet werden
```

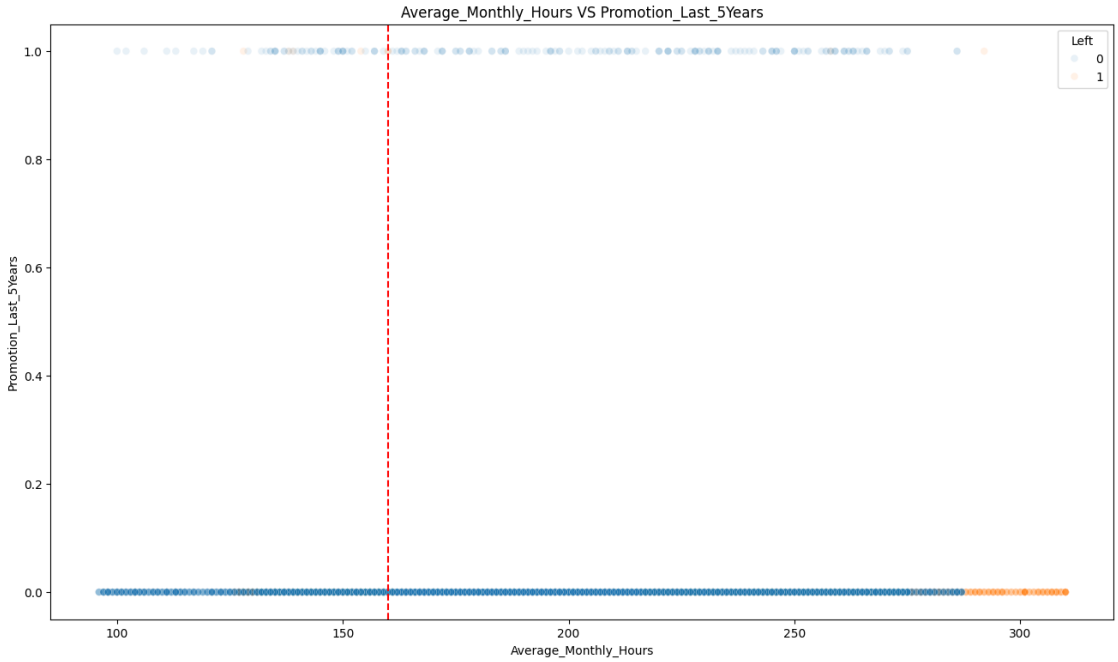
[389]:
```
<matplotlib.lines.Line2D at 0x2ac108fa0>
```



Ergebniss Bei dem ersten Diagramm erkennen wir das es zwei Bereiche gibt, die eine Beziehung zu einander aufweisen. 1. Die erste Gruppe weist eine hohe Bewertung auf, die zeite Gruppe weist eine niedrige Bewertung auf,beide Gruppen verlassen jedich die Firma. Das kann zur Folge haben das die erste Gruppe im Vergleich zur zweiten Gruppe fast doppelt soviel arbeitet.

[404]:
```
#Next, you could examine whether employees who worked very long hours were␣
 ↪promoted in the last five years.
plt.figure(figsize=(16,9))
sns.scatterplot(data=df, x="Average_Monthly_Hours",␣
 ↪y="Promotion_Last_5Years",hue="Left",alpha=0.1)
plt.axvline(x=160,ls= "--",color="red")

plt.title("Average_Monthly_Hours VS Promotion_Last_5Years")
```

[404]:
```
Text(0.5, 1.0, 'Average_Monthly_Hours VS Promotion_Last_5Years')
```



Ergebniss Was wir sehen können: - Von den Mitarbeitern die eine Promotion bekommen haben gibt es nur sehr wenige die das Unternehmen verlassen haben. -Es gibt aller dings sehr viele die keine Promotoin bekommen habeb und Zeitgleich eine sehr hohe Arbeitszeit aufweisen - Es gibt auch

[424]:
```
#Next, you could inspect how the employees who left are distributed across␣
 ↪departments.

# Hier erkennen wir wie viele der Mitarbeiter pro Department das Unternehmen␣
 ↪verlassen haben und wie viele nicht, jedoch bekommen wir als Ergebniss eine␣
 ↪Serie heraus

df.groupby("Left")["Department"].value_counts()

#Hier haben wir die Serie in ein Dataframe umgewandelt, um damit im anschluss␣
 ↪unsere Visualisierung fortsetzen zu können.
a = df.groupby("Left")["Department"].value_counts()


a = a.to_frame()
a
```

[424]:

Left	Department	count
0	sales	2689
	technical	1854

```
        support        1509
        IT              818
        RandD           609
        product_mng     576
        marketing       561
        accounting      512
        hr              488
        management      384
1       sales           550
        technical       390
        support         312
        IT              158
        hr              113
        marketing       112
        product_mng     110
        accounting      109
        RandD            85
        management       52
```

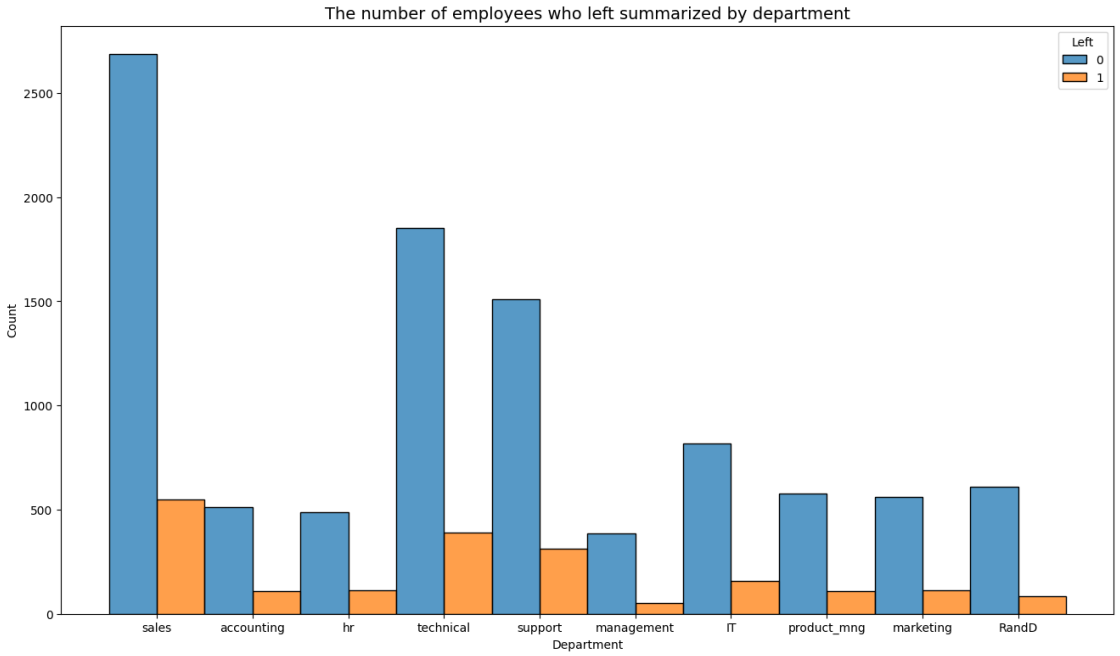
[434]:
```
df["Department"].value_counts()
```

[434]:
```
Department
sales          3239
technical      2244
support        1821
IT              976
RandD           694
product_mng     686
marketing       673
accounting      621
hr              601
management      436
Name: count, dtype: int64
```

Ergebiss Anhand dieser Abfrage erkennen wir ganz deutlich wie viele Mitarbeiter in den jeweiligen Abteilungen das Unternehmen verlassen haben und wie viele das Unternehmen nicht verlassen haben.

[440]:
```
## Create stacked histogram to compare department distribution of employees who␣
 ↪left to that of employees who didn't
plt.figure(figsize=(16,9))

sns.histplot(data=df,x="Department",hue="Left",multiple="dodge")
plt.title("The number of employees who left summarized by␣
 ↪department",fontsize=14)
```

[440]:
```
Text(0.5, 1.0, 'The number of employees who left summarized by department')
```



[448]:
```
#Heatmap
df1 = df.drop(["Salary","Department"],axis=1)
df1
```

[448]:

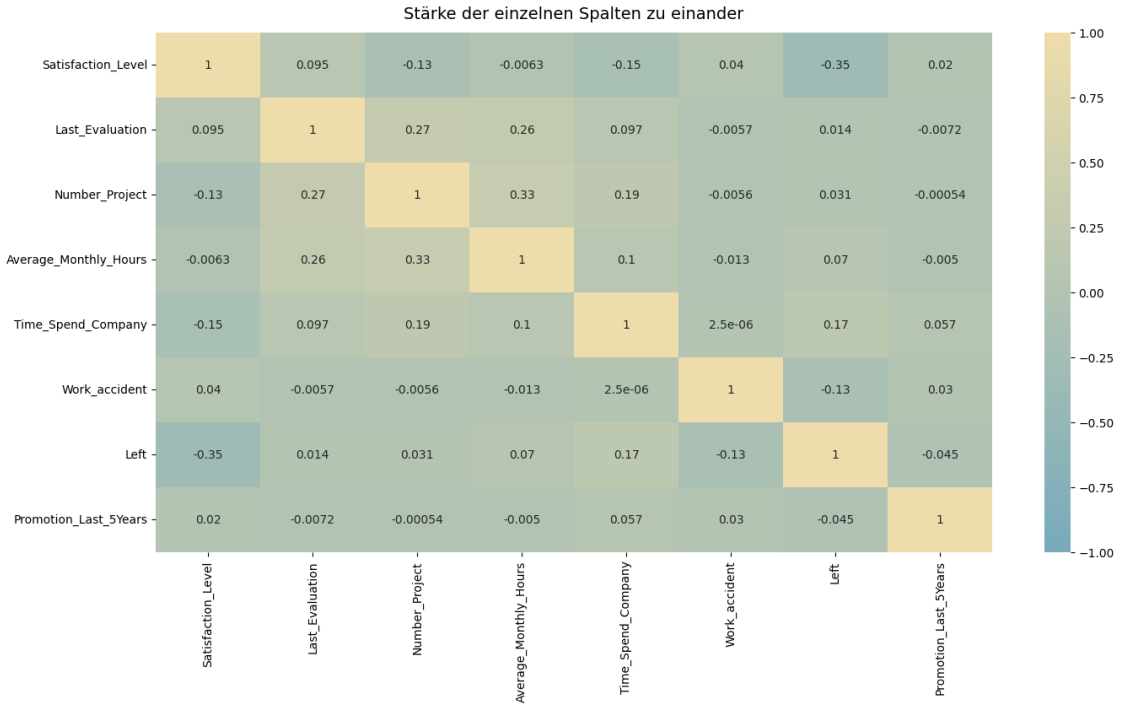
	Satisfaction_Level	Last_Evaluation	Number_Project	\
0	0.38	0.53	2	
1	0.80	0.86	5	
2	0.11	0.88	7	
3	0.72	0.87	5	
4	0.37	0.52	2	
…	…	…	…	
11995	0.90	0.55	3	
11996	0.74	0.95	5	
11997	0.85	0.54	3	
11998	0.33	0.65	3	
11999	0.50	0.73	4	

	Average_Monthly_Hours	Time_Spend_Company	Work_accident	Left	\
0	157	3	0	1	
1	262	6	0	1	
2	272	4	0	1	
3	223	5	0	1	
4	159	3	0	1	
…	…	…	…	…	
11995	259	10	1	0	
11996	266	10	0	0	

```
11997                    185                    10              0      0
11998                    172                    10              0      0
11999                    180                     3              0      0

       Promotion_Last_5Years
0                          0
1                          0
2                          0
3                          0
4                          0
…                          …
11995                      1
11996                      1
11997                      1
11998                      1
11999                      0

[11991 rows x 8 columns]
```

[456]:
```
plt.figure(figsize=(16,8))

sns.heatmap(data=df1.corr(),vmax=1,vmin=-1,annot=True,cmap=sns.
 ↪color_palette("blend:#7AB,#EDA", as_cmap=True))
plt.title("Stärke der einzelnen Spalten zu einander", fontsize=14, pad=12)
```

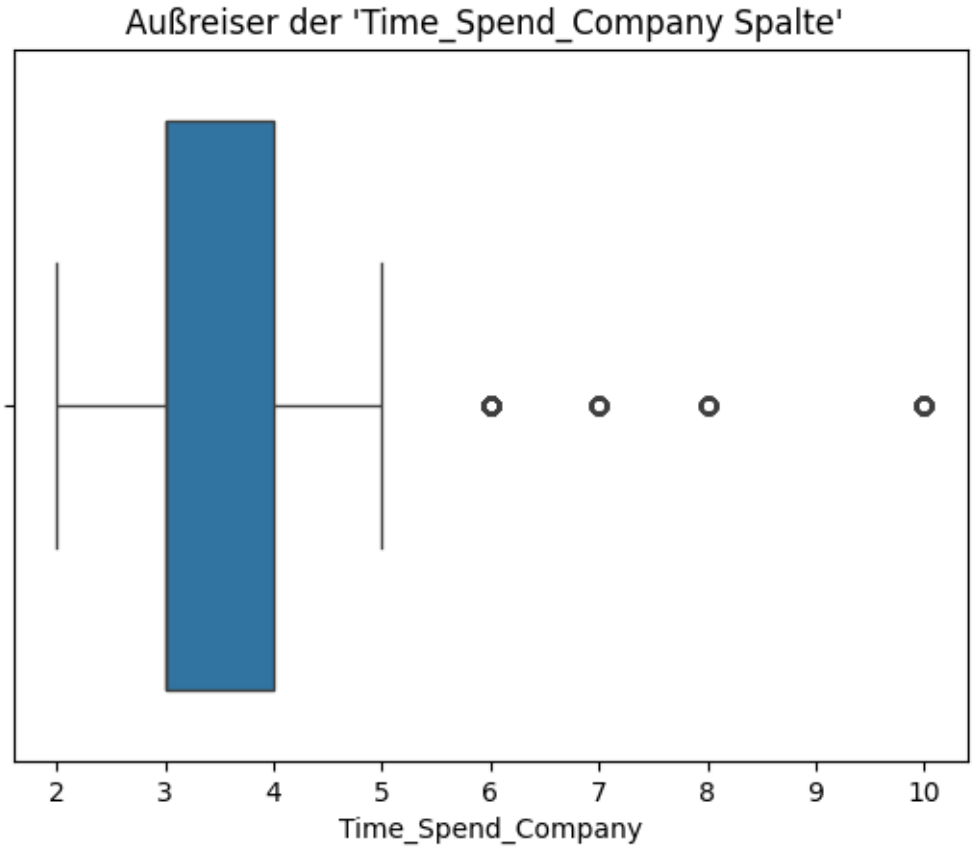
[456]:
```
Text(0.5, 1.0, 'Stärke der einzelnen Spalten zu einander')
```



Ergebniss der Korelationsmatrix Wir erkennen das es eine positive Korrelation von 0,33 zwischen Average Monthly Hours und Number Projekts gibt.

Ausreißer Time_Spend_Company - Was ist die Hauptfrage!! - Was sagen mir die einzellnen Spalten

[ ]:
```
"""Gibt es Sonderfälle was die Arbeitsdauer der Mirarbeiter in der Firma angeht
    ja, bei wem und wieso
    nein, bei wem und wieso"""

#Nutzen Boxplot zur Feststellung von Ausreißern
#palette="PuOr_r"


plt.title("Außreiser der 'Time_Spend_Company Spalte' ")
sns.boxplot(x = df["Time_Spend_Company"])
plt.show()
```



[ ]:
```
""" Anscheined gibt es einige Ausreiser
    - ab wann ist jemand ein Ausreißer
  wer sind SIE und wieso sind die solange dabei """

q1 = df["Time_Spend_Company"].quantile(0.25)

q3 = df["Time_Spend_Company"].quantile(0.75)
print(f"Q1 beträgt {q1} und Q3 beträgt {q3}")
print()

iqa = q3 -q1
print(f"Der Innere Quantile Abstand (iqa) beträgt {iqa}")
print()

"""iqa * 1,5 => das ist der Abstand von q1 nach unten(-) und von q3 nach␣
 ↪oben(+) wo keine Ausreißer vorkommt
    Alles drüber oder drunter sind Ausreißer"""

# WICHTIG Zahlen die mit Komma werden mit NUMPy in Verbindung gebracht

upper_quantile = q3 + (iqa * 1.5)
lower_quantile = q1 - (iqa * 1.5)

print(f"Obere Fläche ist {upper_quantile} und untere Fläche ist␣
 ↪{lower_quantile} alles darunter oder darüber hinaus sind Ausreißer")
print()

#outliers = df1[(df1['tenure'] > upper_limit) | (df1['tenure'] < lower_limit)]

df_outliers = df[(df["Time_Spend_Company"] > upper_quantile) |␣
 ↪(df["Time_Spend_Company"] < lower_quantile)]


#outliers.count()
    #Gibt mir die Spalten mit der Anzahl an Ausreißern

count_of_outliers = len(df_outliers)

print(f"Wir haben insgesamt {count_of_outliers} Ausreißer")
```

```
Q1 beträgt 3.0 und Q3 beträgt 4.0

Der Innere Quantile Abstand (iqa) beträgt 1.0

Obere Fläche ist 5.5 und untere Fläche ist 1.5 alles darunter oder darüber
hinaus sind Ausreißer
```

```
Wir haben insgesamt 824 Ausreißer
```

1. Beobachtung Es gibt nur Ausreißer die länger im UN gearbeitet es gibt keinen der unter 2 Jahre da war 1.Frage: Was hat sie solange gehalten 2.Frage: Wie sind ihre Werte im gegensatz zu Arbeiter die das Unternehmen nach kurzer Zeit verlassen haben

[ ]:
```
#Anzahl der ausreißer pro Department
df_outliers["Department"].value_counts()
```

[ ]:
```
Department
sales          242
technical      130
support        100
management      76
IT              61
marketing       56
product_mng     45
RandD           45
accounting      38
hr              31
Name: count, dtype: int64
```

[ ]:
```
#Durchschnittliche Arbeitszeit eines Mitarbeiters
df_outliers["Average_Monthly_Hours"].mean()
```

[ ]:
```
204.93203883495147
```

[ ]:
```
df["Average_Monthly_Hours"].mean()
```

[ ]:
```
200.4735218080227
```

3 Modeling

Jetzt ist die Modelierung anzugehen. Hier zu muss ein Model entstehen das durch eingabe der Variablen einen Output(Left) herausgibt.

[476]:
```
# Spalten mir categorial data umwandel in numerical Spalten
#Bertriffen sind Deartment(non Ordinary) und Salery(ordinary)

#Duplikat erstellen um die DataFrame bearbeiten zu können
df_enc = df.copy()

#Salery werte zuweisel low<medim<high

#Spalte von categorie zu numerica umändern
df_enc["Salary"] = (
    df_enc["Salary"].
    astype("category").
```

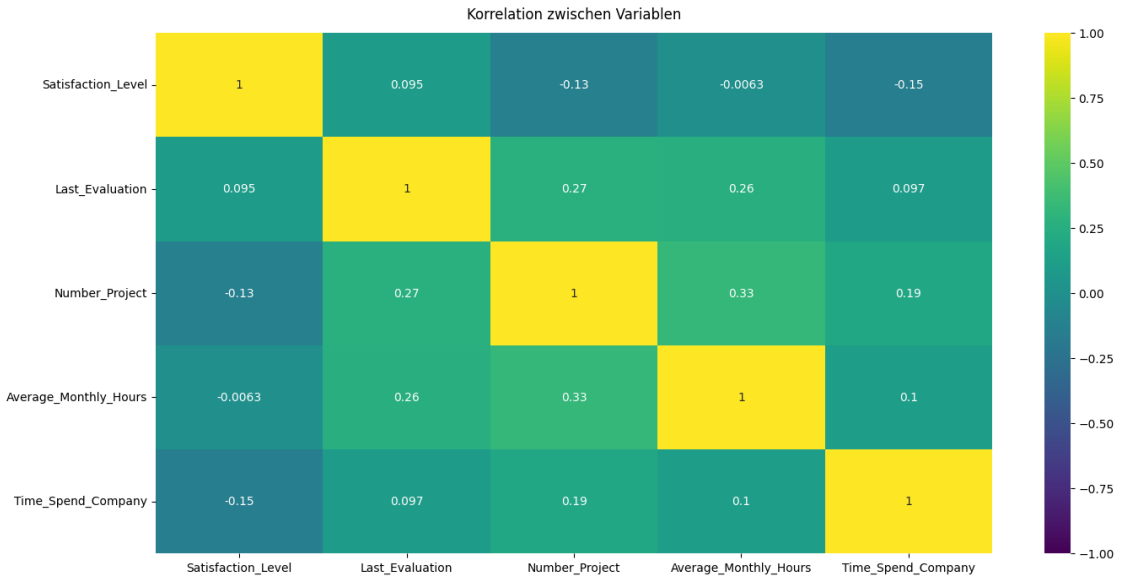
```
    cat.set_categories(["low","medium","high"]).
    cat.codes
)

#Ersetzen in die Dummies durch 0 und q
df_enc = pd.get_dummies(df_enc,drop_first=False)
df_enc = df_enc.replace(False, 0, inplace=False)
df_enc = df_enc.replace(True,1,inplace=False)

df_enc.head()
```

[476]:

	Satisfaction_Level	Last_Evaluation	Number_Project	Average_Monthly_Hours	Time_Spend_Company	Work_accident	Left	Promotion_Last_5Years	Salary	Department_IT	Department_RandD	Department_accounting	Department_hr	Department_management	Department_marketing	Department_product_mng	Department_sales	Department_support	Department_technical
0	0.38	0.53	2	157	3	0	1	0	0	0	0	0	0	0	0	0	1	0	0
1	0.80	0.86	5	262	6	0	1	0	1	0	0	0	0	0	0	0	1	0	0
2	0.11	0.88	7	272	4	0	1	0	1	0	0	0	0	0	0	0	1	0	0
3	0.72	0.87	5	223	5	0	1	0	0	0	0	0	0	0	0	0	1	0	0
4	0.37	0.52	2	159	3	0	1	0	0	0	0	0	0	0	0	0	1	0	0

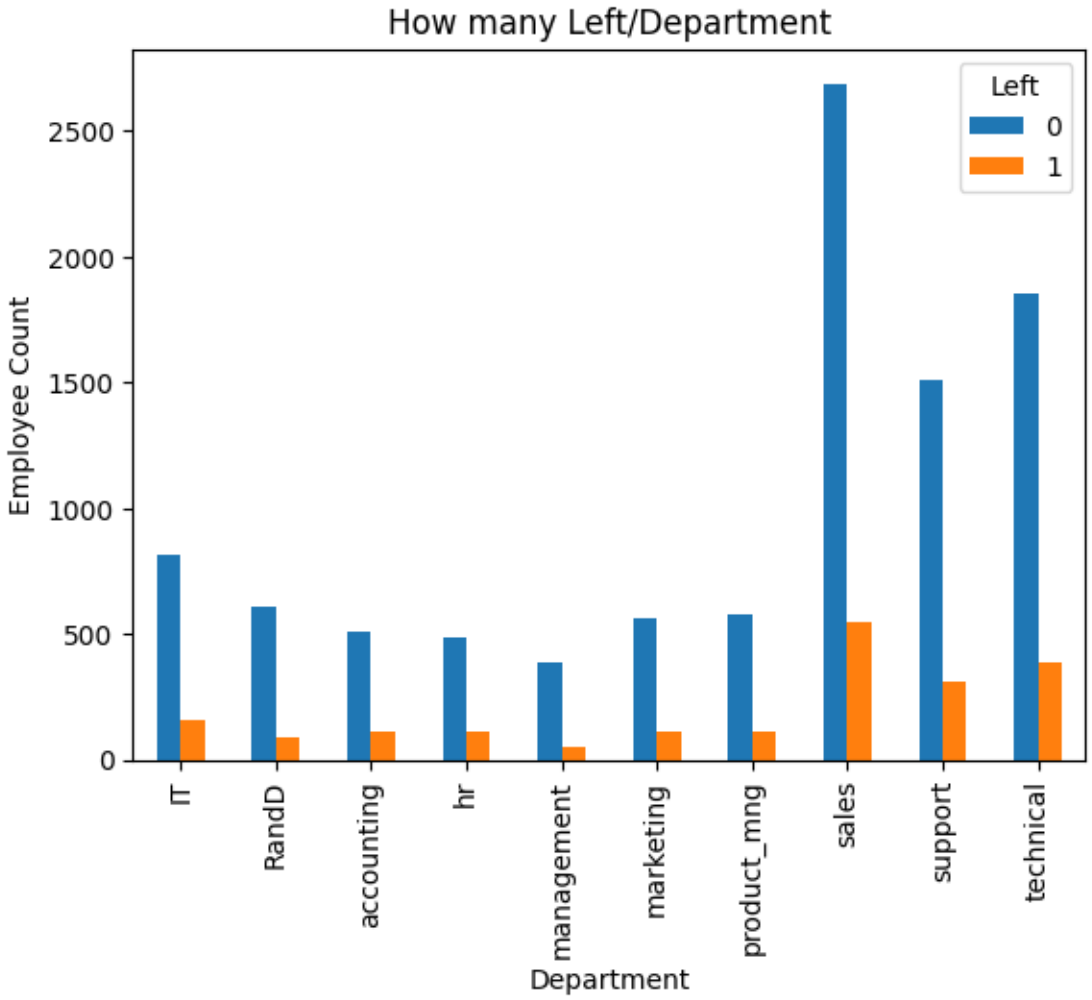
[491]:
```
# Create a heatmap to visualize how correlated variables are
plt.figure(figsize=(16,8))
sns.
 ↪heatmap(data=df_enc[["Satisfaction_Level","Last_Evaluation","Number_Project","Average_Month
 ↪
            corr(),
            vmax=1,
            vmin=-1,
            cmap= sns.color_palette("viridis", as_cmap=True),
            annot=True

            )
plt.title("Korrelation zwischen Variablen", pad=12)
```

[491]:
```
Text(0.5, 1.0, 'Korrelation zwischen Variablen')
```



[498]:
```
# Create a stacked bart plot to visualize number of employees across␣
 ↪department, comparing those who left with those who didn't
# In the legend, 0 (purple color) represents employees who did not leave, 1␣
 ↪(red color) represents employees who left
pd.crosstab(df["Department"],df["Left"]).plot(kind="bar")
plt.title("How many Left/Department")
plt.xlabel("Department")
plt.ylabel("Employee Count")

#Haben eine zusammenfassung der Arbeiter die das Unternehmen verlassen haben␣
 ↪bezogen auf die einzelnen Abteilungen
```

[498]: Text(0, 0.5, 'Employee Count')



[ ]:

[ ]: