Call centre

Sources

The data was created using replit.com and the artificial intelligence ChatGPT, which wrote a Python code to generate random data in the call centre field.

Create by

Abdelrahman Mohamed Ahmed Farouk

The aim of the project

The aim of this project is to analyse and visualize call center data using Power BI to enhance operational efficiency and customer satisfaction. By examining metrics such as call handling time, agent performance, and customer feedback scores, the project seeks to identify key performance indicators and trends. This analysis will support data-driven decision-making and resource allocation, ultimately improving call center performance and service quality.

Code python

```
import pandas as pd
import random

# Function to generate random time strings within a specified range
def generate_time(hours, minutes):
    return f"{hours:02d}:{minutes:02d}"

# List of common agent names
agent_names = [
    "Alice", "Bob", "Charlie", "David", "Eva",
    "Fiona", "George", "Hannah", "Ian", "Jack",
    "Kelly", "Liam", "Mia", "Noah", "Olivia",
```

```
    "Paul", "Quinn", "Rachel", "Steve", "Tina",
    "Ursula", "Victor", "Wendy", "Xander", "Yara", "Zane"
]

# Generate random values for the Calls Table
data = {
    'CallID': range(1, 10001),  # Exactly 10000 Call IDs
    'Date': pd.to_datetime([f"2023-{random.randint(1, 12):02d}-{random.randint(1, 28):02d}" for _ in range(10000)]),
    'Time': [generate_time(random.randint(0, 23), random.randint(0, 59)) for _ in range(10000)],
    'AgentID': [f"{i:02d}" for i in random.choices(range(1, 100), k=10000)],  # Format AgentIDs as two digits
    'CustomerID': random.choices(range(100000, 200000), k=10000),
    'HandlingTime': random.choices(range(2, 15), k=10000),
    'HoldTime': random.choices(range(0, 3), k=10000),
    'AvailableTime': random.choices(range(30, 60), k=10000),
    'ProductiveTime': random.choices(range(1, 8), k=10000),
    'SLA': random.choices(range(1, 31), k=10000),
    'ACW': random.choices(range(0, 4), k=10000),
    'Resolution': random.choices(['Yes', 'No'], k=10000, weights=[0.8, 0.2])
}

# Add ReasonOfCall with predefined reasons
reasons = ['Billing Inquiry', 'Technical Support', 'General Inquiry', 'Feedback', 'Complaint']
data['ReasonOfCall'] = random.choices(reasons, k=10000)

# Adjust VOC scores based on Resolution
data['VOC Score'] = [
    random.choices(range(1, 6), weights=[0.05, 0.1, 0.15, 0.3, 0.4])[0] if res == 'Yes'
    else random.choices(range(1, 6), weights=[0.4, 0.25, 0.15, 0.1, 0.1])[0]
    for res in data['Resolution']
]
```

```
# Generate random Quality Scores ensuring a higher average
data['Quality Score'] = [random.choices(range(1, 6), weights=[0.1, 0.15, 0.25, 0.3, 0.2])[0] for _ in range(10000)]

# Create the Calls Table DataFrame
calls_df = pd.DataFrame(data)
calls_df['CallDuration'] = calls_df['HandlingTime'] + calls_df['HoldTime']

# Generate random values for the Agents Table
agents_data = {
    'AgentID': [f"{i:02d}" for i in range(1, 100)],  # 99 agents with IDs from 01 to 99
    'AgentName': random.choices(agent_names, k=99),
    'PaidHoursOnSite': random.choices(range(375, 400), k=99),
    'ScheduledHours': random.choices(range(380, 420), k=99),
    'MissingHours': random.choices(range(0, 10), k=99),
    'TotalOffQueueTime': [random.choice(range(50, 100)) for _ in range(99)]  # Added TotalOffQueueTime
}

# Create the Agents Table DataFrame
agents_df = pd.DataFrame(agents_data)

# Create a new table for NumberOfHeadsToReplace and TotalHC based on unique call dates
unique_dates = calls_df['Date'].unique()
replacement_data = {
    'Date': pd.to_datetime(unique_dates),  # Unique dates from Calls Table
    'NumberOfHeadsToReplace': random.choices(range(0, 5), k=len(unique_dates)),
    'TotalHC': random.choices(range(80, 100), k=len(unique_dates))
}
capacity_df = pd.DataFrame(replacement_data)
```

```
# Save the files locally
calls_df.to_excel('Calls_Data.xlsx', index=False)
agents_df.to_excel('Agents_Data.xlsx', index=False)
capacity_df.to_excel('Capacity_Data.xlsx', index=False)

# Output file paths
print("Files saved at:")
print("Calls_Data.xlsx")
print("Agents_Data.xlsx")
print("Capacity_Data.xlsx")
```

Summary

3 files

.xlsx

Calls_data
10,000 rows

Agents_data
100 rows

Capacity_data
365 rows

About Dataset

Agents_data

Header	Description	Data Type
AgentID	Unique identifier for each agent	Integer
AgentName	Name of the agent	String
PaidHoursOnSite	Total hours (in minutes) the agent was paid for working on-site	Integer
ScheduledHours	Total hours (in minutes) the agent was scheduled to work	Integer
MissingHours	Total hours (in minutes) the agent was absent or not scheduled	Integer
TotalOffQueueTime	Total time (in minutes) the agent spent off queue during their shifts	Integer

Capacity_data

Header	Description	Data Type
Date	Each day date 2023	Date (YYYY-MM-DD)
NumberOfHeadsToReplace	Number of agents that need to be replaced due to various reasons	Integer
TotalOffQueueTime	Total time (in minutes) the agent spent off queue during their shifts	Integer

Calls_data

Header	Description	Data Type
CallID	Unique identifier for each call	Integer
Date	The date when the call took place	Date (YYYY-MM-DD)
Time	The time when the call occurred	time
AgentID	Unique identifier for the agent handling the call	Integer
CustomerID	Unique identifier for the customer who made the call	Integer
HandlingTime	Duration (in minutes) the agent spent handling the call	Integer
HoldTime	Duration (in minutes) the customer was on hold during the call	Integer
AvailableTime	Duration (in minutes) the agent was available during the call	Integer
ProductiveTime	Duration (in minutes) spent on productive activities during the call	Integer
SLA	Service Level Agreement time (in minutes) expected for the call	Integer
ACW	After Call Work time (in minutes) the agent spent after the call	Integer
Resolution	Indicates whether the call was resolved ("Yes" or "No")	String
ReasonOfCall	The reason for the call (e.g., Technical Support, Complaint)	String
VOC Score	Voice of the Customer score (1 to 5 rating) reflecting customer feedback CSAT satisfied 4-5 DSAT dis-satisfied 1-3	Integer (1-5)
Quality Score	Quality score (1 to 5 rating) reflecting the quality of the service	Integer (1-5)
CallDuration	Total duration of the call (HandlingTime + HoldTime)	Integer

KPIs Calculation

Utilization

Utilization is the percentage of paid time that the agent is either performing productive work or available to handle a customer.

Total Handling Time + Available Time/Paid hours on site.

Occupancy

Occupancy is the percentage of time that the agent is engaged in productive work out of the total time he is ready to do productive work.

Total Handling Time /Available Time +Productive time

Absenteeism

Absenteeism is the total number of lost planned hours that should have been covered during a certain period.

Missing hour/scheduled hours

Attrition

Attrition is the percentage of lost heads that needs to be replaced.

Number of heads to replace /total HC

Shrinkage

On site shrinkage: Offline activities that prevent the agent from being productive.

Off site shrinkage: Absenteeism & Attrition

Total off queue time /Paid hours on site

AHT

Average handling time, AHT is the average time it takes to handle a real time transaction including any work carried out after the customer gets disconnected.

Total talk time+ ACW+Hold/Total number of calls

SLA

SLA is the percentage of calls answered within an established threshold

Numbers of calls answered with threshold/Total number of recieved calls

VOC

VOC is measuring your customers' satisfaction rate with the service you're providing.

C-sat = Total positive surveys / Total number of surveys.

D-sat = Total negative surveys / Total number of surveys.

NPS= Promotors-Detractors / Total number of surveys.

Quality

Quality is measuring how compliant you are with the client's process and procedures, measuring any deviation based on three different critical errors.

Number of calls without critical error /Total number of monitored calls.

Conversion

Conversion is to track the sales rate "the percentage of the calls with a sale" or to track the sales volume "Units Sold".

Number of calls with a success sale /Total number of answered calls