Tidyverse

1. Conhecendo os dados

Importar e visualizar dados

Carregando o Tidyverse

```
install.packages("tidyverse")
```

```
Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
(as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
```

```
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

Carregando os dados

```r
transacoes_milhas <- read_csv("transacoes_milhas.csv")
```

```
Rows: 1000 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
dbl  (3): id_cliente, milhas_vendidas, valor_recebido
date (1): data_venda_milhas

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
transacoes_passagens <- read_csv("transacoes_passagens.csv")
```

```
Rows: 1000 Columns: 6
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (1): classe_voo
dbl  (4): id_cliente, milhas_utilizadas, valor_pago, numero_voo
date (1): data_transacao

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
clientes <- read_csv("clientes.csv")
```

```
Rows: 1000 Columns: 5
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (1): status_fidelidade
dbl  (3): id_cliente, milhas_acumuladas, gasto_total
date (1): data_nascimento

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Analisando dados

```r
transacoes_milhas %>% head()
```

```
# A tibble: 6 x 4
  id_cliente data_venda_milhas milhas_vendidas valor_recebido
       <dbl> <date>                      <dbl>          <dbl>
1        350 2023-02-23                   4131           537.
2        109 2022-09-26                   6525           114.
3        376 2022-06-12                   9955           467.
4        443 2022-12-17                   5913          1490.
5        647 2023-10-06                    284          1908.
6        818 2023-05-29                   4225          1339.
```

```r
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 6
  id_cliente data_transacao milhas_utilizadas valor_pago classe_voo   numero_voo
       <dbl> <date>                     <dbl>      <dbl> <chr>             <dbl>
1        507 2023-01-30                  9837       424. Economica          3705
2         86 2023-03-07                  6067       514. Economica          9504
3        589 2022-02-11                  2893      1320. Primeira Cl~       1695
4        774 2022-09-23                  2274      1000. Economica          2963
5        705 2023-06-08                  6043      1607. Economica          9190
6        316 2023-05-13                  1798      1803. Executiva          6242
```

```r
transacoes_milhas %>% glimpse()
```

```
Rows: 1,000
Columns: 4
$ id_cliente        <dbl> 350, 109, 376, 443, 647, 818, 32, 770, 707, 842, 170~
$ data_venda_milhas <date> 2023-02-23, 2022-09-26, 2022-06-12, 2022-12-17, 202~
$ milhas_vendidas   <dbl> 4131, 6525, 9955, 5913, 284, 4225, 2854, 6127, 5091,~
$ valor_recebido    <dbl> 536.88, 113.70, 466.73, 1489.56, 1908.38, 1339.10, 1~
```

```r
clientes %>% glimpse()
```

```
Rows: 1,000
Columns: 5
$ id_cliente        <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 1~
```

```
$ data_nascimento   <date> 1971-01-09, 1996-08-11, 1994-07-18, 2003-03-08, 196~
$ status_fidelidade <chr> "Diamante", "Diamante", "Diamante", "Prata", "Diaman~
$ milhas_acumuladas <dbl> 2833, 8864, 917, 8773, 1492, 6270, 2583, 8263, 8914,~
$ gasto_total       <dbl> 1906.69, 493.58, 519.16, 1256.60, 410.37, 250.93, 92~
```

Seleção e filtragem de dados

Retomando os dados

```
transacoes_milhas %>% glimpse()
```

```
Rows: 1,000
Columns: 4
$ id_cliente        <dbl> 350, 109, 376, 443, 647, 818, 32, 770, 707, 842, 170~
$ data_venda_milhas <date> 2023-02-23, 2022-09-26, 2022-06-12, 2022-12-17, 202~
$ milhas_vendidas   <dbl> 4131, 6525, 9955, 5913, 284, 4225, 2854, 6127, 5091,~
$ valor_recebido    <dbl> 536.88, 113.70, 466.73, 1489.56, 1908.38, 1339.10, 1~
```

```
transacoes_passagens %>% glimpse()
```

```
Rows: 1,000
Columns: 6
$ id_cliente        <dbl> 507, 86, 589, 774, 705, 316, 801, 149, 11, 154, 363,~
$ data_transacao    <date> 2023-01-30, 2023-03-07, 2022-02-11, 2022-09-23, 202~
$ milhas_utilizadas <dbl> 9837, 6067, 2893, 2274, 6043, 1798, 4354, 6976, 6344~
$ valor_pago        <dbl> 424.31, 513.52, 1320.34, 1000.44, 1607.35, 1802.67, ~
$ classe_voo        <chr> "Economica", "Economica", "Primeira Classe", "Econom~
$ numero_voo        <dbl> 3705, 9504, 1695, 2963, 9190, 6242, 2415, 6747, 3509~
```

Selecionando colunas

```
transacoes_passagens <- transacoes_passagens %>% select(id_cliente, data_transacao, milhas_u
```

```
transacoes_passagens %>% glimpse()
```

```
Rows: 1,000
Columns: 4
$ id_cliente        <dbl> 507, 86, 589, 774, 705, 316, 801, 149, 11, 154, 363,~
$ data_transacao    <date> 2023-01-30, 2023-03-07, 2022-02-11, 2022-09-23, 202~
$ milhas_utilizadas <dbl> 9837, 6067, 2893, 2274, 6043, 1798, 4354, 6976, 6344~
$ valor_pago        <dbl> 424.31, 513.52, 1320.34, 1000.44, 1607.35, 1802.67, ~
```

Selecionando períodos

```
transacoes_milhas <- transacoes_milhas %>% filter(data_venda_milhas >= '2022-01-01' & data_ve
```

```
transacoes_milhas %>% glimpse()
```

```
Rows: 485
Columns: 4
$ id_cliente        <dbl> 109, 376, 443, 32, 842, 609, 391, 600, 101, 761, 959~
$ data_venda_milhas <date> 2022-09-26, 2022-06-12, 2022-12-17, 2022-12-06, 202~
$ milhas_vendidas   <dbl> 6525, 9955, 5913, 2854, 5686, 937, 1190, 8295, 635, ~
$ valor_recebido    <dbl> 113.70, 466.73, 1489.56, 1983.85, 191.69, 183.50, 47~
```

```
transacoes_passagens <- transacoes_passagens %>% filter(data_transacao >= '2022-01-01' & data
```

```
transacoes_passagens %>% glimpse()
```

```
Rows: 502
Columns: 4
$ id_cliente        <dbl> 589, 774, 149, 620, 96, 449, 693, 242, 358, 317, 633~
$ data_transacao    <date> 2022-02-11, 2022-09-23, 2022-11-07, 2022-08-12, 202~
$ milhas_utilizadas <dbl> 2893, 2274, 6976, 6479, 9444, 8102, 5335, 7139, 7280~
$ valor_pago        <dbl> 1320.34, 1000.44, 309.56, 1869.74, 267.63, 268.38, 1~
```

Criar e modificar colunas com mutate

Calculando o custo por milha

```
transacoes_passagens <- transacoes_passagens %>% mutate(custo_por_milha = valor_pago / milhas
```

```
transacoes_passagens %>% glimpse()
```

```
Rows: 502
Columns: 5
$ id_cliente        <dbl> 589, 774, 149, 620, 96, 449, 693, 242, 358, 317, 633~
$ data_transacao    <date> 2022-02-11, 2022-09-23, 2022-11-07, 2022-08-12, 202~
$ milhas_utilizadas <dbl> 2893, 2274, 6976, 6479, 9444, 8102, 5335, 7139, 7280~
$ valor_pago        <dbl> 1320.34, 1000.44, 309.56, 1869.74, 267.63, 268.38, 1~
$ custo_por_milha   <dbl> 0.45639129, 0.43994723, 0.04437500, 0.28858466, 0.02~
```

Calculando a média de milhas

```
media_milhas <- transacoes_passagens %>% group_by(id_cliente) %>% summarize(media_milhas = me
```

```
media_milhas %>% head()
```

```
# A tibble: 6 x 2
  id_cliente media_milhas
       <dbl>        <dbl>
1          3        1023
2          7        7074.
3         10        6537
4         12        9312
5         13        4123
6         15        7541
```

2. Organizando, combinando e filtrando dataframes

Ordenar compras de passagens por data com arrange

Ordenando dataframes

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 5
  id_cliente data_transacao milhas_utilizadas valor_pago custo_por_milha
       <dbl> <date>                     <dbl>      <dbl>           <dbl>
1        589 2022-02-11                  2893      1320.          0.456
2        774 2022-09-23                  2274      1000.          0.440
3        149 2022-11-07                  6976       310.          0.0444
4        620 2022-08-12                  6479      1870.          0.289
5         96 2022-05-23                  9444       268.          0.0283
6        449 2022-10-16                  8102       268.          0.0331
```

```
transacoes_passagens <- transacoes_passagens %>% arrange(data_transacao)
```

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 5
  id_cliente data_transacao milhas_utilizadas valor_pago custo_por_milha
       <dbl> <date>                     <dbl>      <dbl>           <dbl>
1        967 2022-01-01                  1198      1217.           1.02
2        747 2022-01-01                  8451      1064.          0.126
3        237 2022-01-01                  7996      1685.          0.211
4        504 2022-01-02                  6143       693.          0.113
5        840 2022-01-02                  5267      1497.          0.284
6        396 2022-01-02                  1929       760.          0.394
```

Unir dados de compras e vendas com left_join

Unindo dataframes

```
transacoes_passagens <- transacoes_passagens %>% left_join(clientes, by="id_cliente")
```

Exibindo o dataframe

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 9
  id_cliente data_transacao milhas_utilizadas valor_pago custo_por_milha
       <dbl> <date>                     <dbl>      <dbl>           <dbl>
```

```
1        967 2022-01-01                  1198      1217.           1.02
2        747 2022-01-01                  8451      1064.           0.126
3        237 2022-01-01                  7996      1685.           0.211
4        504 2022-01-02                  6143       693.           0.113
5        840 2022-01-02                  5267      1497.           0.284
6        396 2022-01-02                  1929       760.           0.394
# i 4 more variables: data_nascimento <date>, status_fidelidade <chr>,
#   milhas_acumuladas <dbl>, gasto_total <dbl>
```

Renomear colunas

Função rename()

Modificando a apresentação dos dados

```
transacoes_passagens <- transacoes_passagens %>% rename(
  Data_Compra_Passagem = data_transacao,
  Milhas_Utilizadas = milhas_utilizadas,
  Valor_Pago = valor_pago,
  Custo_Por_Milha = custo_por_milha,
  Data_Nascimento = data_nascimento,
  Status_Fidelidade = status_fidelidade,
  Milhas_Acumuladas = milhas_acumuladas,
  Gasto_Total = gasto_total
)
```

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 9
  id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
       <dbl> <date>                           <dbl>      <dbl>           <dbl>
1        967 2022-01-01                        1198      1217.           1.02
2        747 2022-01-01                        8451      1064.           0.126
3        237 2022-01-01                        7996      1685.           0.211
4        504 2022-01-02                        6143       693.           0.113
5        840 2022-01-02                        5267      1497.           0.284
6        396 2022-01-02                        1929       760.           0.394
# i 4 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>
```

Quantidade de passagens por cliente

```
transacoes_passagens %>% count(id_cliente)
```

```
# A tibble: 394 x 2
   id_cliente     n
        <dbl> <int>
 1          3     1
 2          7     2
 3         10     1
 4         12     1
 5         13     1
 6         15     1
 7         23     1
 8         26     2
 9         28     1
10         29     1
# i 384 more rows
```

```
transacoes_passagens %>% count(id_cliente) %>% rename(Total_Compras=n) %>% arrange(desc(Total
```

```
# A tibble: 6 x 2
  id_cliente Total_Compras
       <dbl>         <int>
1        586             5
2        544             4
3        861             4
4        175             3
5        207             3
6        239             3
```

3. Aplicando técnicas de transformação de dados

Contar o número de compras por cliente

Selecionando apenas colunas de interesse

```r
dados_milhas <- transacoes_passagens %>%
  select(id_cliente, Milhas_Utilizadas)
```

Calculando o total de milhas usadas por cliente

```r
total_milhas_por_cliente <- dados_milhas %>%
  group_by(id_cliente) %>%
  summarise(total_milhas = sum(Milhas_Utilizadas, na.rm =TRUE))
```

Exibindo clientes que mais usaram milhas

```r
total_milhas_por_cliente %>%
  arrange(desc(total_milhas)) %>%
  head()
```

```
# A tibble: 6 x 2
  id_cliente total_milhas
       <dbl>        <dbl>
1        544        28726
2        572        22209
3        890        19566
4         96        19047
5        945        18514
6         32        18126
```

Criação de novas colunas e classificação

Criando uma nova coluna de custo por milha

```r
transacoes_passagens <- transacoes_passagens %>% mutate(categoria_cliente = case_when(
  Milhas_Utilizadas < 2000 ~ 'Baixo',
  Milhas_Utilizadas < 5000 ~ 'Médio',
TRUE ~ 'Alto'
))
```

Verificando o resultado

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 10
  id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
       <dbl> <date>                           <dbl>      <dbl>           <dbl>
1        967 2022-01-01                        1198      1217.           1.02
2        747 2022-01-01                        8451      1064.           0.126
3        237 2022-01-01                        7996      1685.           0.211
4        504 2022-01-02                        6143       693.           0.113
5        840 2022-01-02                        5267      1497.           0.284
6        396 2022-01-02                        1929       760.           0.394
# i 5 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>, categoria_cliente <chr>
```

4. Entendendo o comportamento do cliente

Diferença de milhas e previsão

```
transacoes_passagens <- transacoes_passagens %>%
  arrange(id_cliente, Data_Compra_Passagem) %>%
  group_by(id_cliente) %>%
  mutate(
    diferenca_milhas = Milhas_Utilizadas - lag(Milhas_Utilizadas),
    diferenca_valor_futuro = lead(Milhas_Utilizadas )
  ) %>%
  ungroup()
```

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 12
  id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
       <dbl> <date>                           <dbl>      <dbl>           <dbl>
1          3 2022-03-24                        1023      1588.          1.55
2          7 2022-09-16                        9449      1906.          0.202
3          7 2022-12-07                        4700      1286.          0.274
```

```
4         10 2022-06-07                          6537       469.          0.0718
5         12 2022-10-29                          9312       570.          0.0612
6         13 2022-09-07                          4123       240.          0.0583
# i 7 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>, categoria_cliente <chr>,
#   diferenca_milhas <dbl>, diferenca_valor_futuro <dbl>
```

Filtragem e tratamento de valores ausentes

```r
any(is.na(transacoes_passagens))
```

```
[1] TRUE
```

```r
sum(is.na(transacoes_passagens))
```

```
[1] 788
```

```r
num_clientes <- transacoes_passagens %>%
  summarise(num_clientes = n_distinct(id_cliente))
num_clientes
```

```
# A tibble: 1 x 1
  num_clientes
         <int>
1          394
```

```r
transacoes_passagens <- transacoes_passagens %>%
  mutate(
    diferenca_milhas = coalesce(diferenca_milhas, 0),
    diferenca_valor_futuro = coalesce(diferenca_valor_futuro, 0)
  )
```

```r
any(is.na(transacoes_passagens))
```

```
[1] FALSE
```

```r
transacoes_passagens %>% arrange(desc(diferenca_milhas)) %>% head(10)
```

```
# A tibble: 10 x 12
   id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
        <dbl> <date>                           <dbl>      <dbl>           <dbl>
 1        625 2022-12-30                        7566      1891.          0.250
 2        961 2022-12-30                        7215       594.          0.0824
 3        151 2022-08-28                        8923       689.          0.0772
 4        820 2022-11-09                        9658       473.          0.0490
 5        807 2022-10-24                        7239       361.          0.0499
 6        370 2022-09-23                        8858       884.          0.0999
 7        586 2022-11-20                        9478      1070.          0.113
 8        945 2022-12-19                        7559      1442.          0.191
 9        332 2022-09-13                        8010      1669.          0.208
10        524 2022-05-01                        7567      1649.          0.218
# i 7 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>, categoria_cliente <chr>,
#   diferenca_milhas <dbl>, diferenca_valor_futuro <dbl>
```

```r
transacoes_passagens %>% arrange(desc(diferenca_valor_futuro)) %>% head(10)
```

```
# A tibble: 10 x 12
   id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
        <dbl> <date>                           <dbl>      <dbl>           <dbl>
 1        912 2022-05-11                        4572       494.          0.108
 2        685 2022-06-19                        4529      1613.          0.356
 3        371 2022-03-03                        5410       144.          0.0266
 4        820 2022-07-06                        2810      1540.          0.548
 5        890 2022-02-01                        9951       638.          0.0641
 6        586 2022-09-06                        3256       358.          0.110
 7         96 2022-03-14                        9603      1917.          0.200
 8        548 2022-02-19                        6482       687.          0.106
 9        144 2022-05-08                        6847      1435.          0.210
10        583 2022-01-29                        5653       435.          0.0770
# i 7 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>, categoria_cliente <chr>,
#   diferenca_milhas <dbl>, diferenca_valor_futuro <dbl>
```

5. Impulsionando o crescimento com análise de clientes

Classificar clientes com case_when e recode

Ajuste de nomenclatura para apresentação dos resultados

```
transacoes_passagens <- transacoes_passagens %>%
    mutate(categoria_cliente =
        recode (categoria_cliente,
        "Baixo" = "Iniciante",
        "Médio" = "Intermediário",
        "Alto" = "Avançado"
        )
    )
```

```
transacoes_passagens %>% head()
```

```
# A tibble: 6 x 12
  id_cliente Data_Compra_Passagem Milhas_Utilizadas Valor_Pago Custo_Por_Milha
       <dbl> <date>                           <dbl>      <dbl>           <dbl>
1          3 2022-03-24                        1023      1588.          1.55
2          7 2022-09-16                        9449      1906.          0.202
3          7 2022-12-07                        4700      1286.          0.274
4         10 2022-06-07                        6537       469.          0.0718
5         12 2022-10-29                        9312       570.          0.0612
6         13 2022-09-07                        4123       240.          0.0583
# i 7 more variables: Data_Nascimento <date>, Status_Fidelidade <chr>,
#   Milhas_Acumuladas <dbl>, Gasto_Total <dbl>, categoria_cliente <chr>,
#   diferenca_milhas <dbl>, diferenca_valor_futuro <dbl>
```

Transformar colunas em linhas com pivot_longer

Criando o dataframe longo

```
transacoes_passagens %>% select (id_cliente, Milhas_Utilizadas, Milhas_Acumuladas) %>% head
```

```
# A tibble: 6 x 3
  id_cliente Milhas_Utilizadas Milhas_Acumuladas
```

```
       <dbl>             <dbl>             <dbl>
1          3              1023               917
2          7              9449              2583
3          7              4700              2583
4         10              6537              3829
5         12              9312              8857
6         13              4123              2228
```

```
df_long <- transacoes_passagens %>%
            pivot_longer(
            cols = starts_with("Milhas"),
            names_to = "tipo_milhas",
            values_to = "quantidade"
            )%>%
select(id_cliente, tipo_milhas, quantidade)
df_long %>% head()
```

```
# A tibble: 6 x 3
  id_cliente tipo_milhas       quantidade
       <dbl> <chr>                  <dbl>
1          3 Milhas_Utilizadas       1023
2          3 Milhas_Acumuladas        917
3          7 Milhas_Utilizadas       9449
4          7 Milhas_Acumuladas       2583
5          7 Milhas_Utilizadas       4700
6          7 Milhas_Acumuladas       2583
```

Agrupando dados e calculando métricas

```
df_long %>%
    group_by(tipo_milhas) %>%
    summarise(
    media = mean(quantidade),
    mediana = median(quantidade),
    desvio_padrao = sd(quantidade),
)
```

```
# A tibble: 2 x 4
  tipo_milhas       media mediana desvio_padrao
  <chr>             <dbl>   <dbl>         <dbl>
```

```
1 Milhas_Acumuladas 5029.   4926.         2917.
2 Milhas_Utilizadas 5113.   5260.         2793.
```