

Para não perder nossas informações

Vimos que podemos gerar relatórios de frequências especiais, chamados de **frequências cruzadas**, onde vemos como as classes de duas variáveis diferentes interagem entre si. Para fazermos essa tabela usando o `PROC FREQ`, basta colocarmos um asterisco (*) entre as duas variáveis que desejamos visualizar na nossa frequência cruzada, da seguinte forma:

```
PROC FREQ
  data = alura.cadastro_produto;
  table genero*plataforma;
RUN;
```

Vimos que o padrão do SAS é gerar uma tabela de frequência cruzada em formato matricial com as seguintes informações por célula:

- Frequência (quantidade) da célula.
- Percentual da célula (em relação à tabela toda).
- Percentual que a célula representa na linha.
- Percentual que a célula representa na coluna.

Mas o `PROC FREQ` também possui opções que nos permitem suprimir algumas das informações para que elas não sejam mostradas na tabela. Por exemplo:

- `norow` suprime as informações relacionadas à linha (se estas fossem mostradas).
- `nocol` suprime as informações relacionadas à coluna (se estas fossem mostradas).
- `nofreq` suprime as informações de frequência (inclusive a acumulada).
- `nopercent` suprime as informações de percentual (inclusive o acumulado).

Assim, se desejarmos gerar a mesma frequência cruzada acima, mas mostrando apenas a informação de frequência de cada célula podemos usar o código abaixo:

```
PROC FREQ
  data=teste;
  table genero*lancamento
  /norow nocol nopercent;
RUN;
```

Também podemos, dentro de um único `PROC FREQ`, passar mais de um comando `table`, se quisermos fazer listagens de frequências de uma mesma base, mas usando diferentes opções, como no exemplo abaixo:

```
PROC FREQ
  data=alura.cadastro_produto nlevels;
  table nome;
  table nome*genero
  / list;
RUN;
```

Mas a nossa base ainda está sendo salva na *work*, uma biblioteca automaticamente gerada pelo SAS onde são colocadas todas as bases que são criadas fora de uma biblioteca especificada pelo usuário. Essa biblioteca é temporária, no sentido que sempre que eu encerro minha sessão no SAS, fechando o programa, essa biblioteca “deixa de existir”, e tudo o que estava nela é apagado. Ao iniciar uma nova sessão, uma nova work é gerada.

Não queremos perder a nossa base! Por isso precisamos salvá-la no nosso diretório do projeto da AluraPlay. Salve a base onde você criou a variável `lançamento` na biblioteca `alura`, usando um *data step*, mas aproveite para, ao mesmo tempo, renomear a variável `lançamento` para `flag_lançamento` e, usando o comando `label`, definir os rótulos da variável `Genero` como “Gênero” e da variável `lançamento` como o texto “Marca 1 para jogos que são lançamento e 0 caso contrário”.