

# Pass-Through Câmbio Inflação

## Temas:

- Utilização de banco de dados relacionais
  - Consulta a banco de dados
  - Importação de dados
- Estimar modelo VAR – Impulso Resposta

## Bibliografia:

*Analysis of Integrated and Cointegrated Time Series with R*

Bernhard Pfaff

Springer – User R!

# Modelagem Quantitativa do Problema

## 1. Entender o problema

Referências bibliográficas: Qual a relevância da bibliografia (citações, jornal de publicação, tipo de bibliografia)

Quais os modelos utilizados? Em que contexto? Quais os dados ?

 **MODELO: VAR / Impulso Resposta**

## 2. Entender os dados:

**Fonte Única:** Bacen **Periodicidade:** Mensal

- Como obter os dados?
- Procedimento recorrente ou uma única vez? Automatizar?
- Forma de Armazenamento: Planilhas, Arquivos txt, Base SQL
- Análises descritivas e verificação da qualidade do dado
- Desenvolvimento de scripts especialistas

 ***Data Clearing e Data Preparation***

## 3. Desenvolvimento do Modelo

## 4. Verificação e Validação

# Pass-Through Câmbio Inflação

## Observação:

- Por motivos didáticos vamos abordar o problema utilizando um banco de dados SQL
- O problema poderia ter sido abordado utilizando planilhas ou arquivos texto.
- Vamos iniciar com consultas na base e em seguida vamos automatizar a inserção de dados na base.

# Consultas a Bancos de Dados com RODBC/ACCESS

Como acessar uma base de dado Access utilizando o *package* RODBC:

```
## --- Bibliotecas R
library(RODBC);

## --- Programa Principal

## - Carrega os dados
databasefile = "../database/<nome_do_arquivo|access>"

## Abre a conexão com o banco de dados
conn = odbcConnectAccess2007(databasefile);

## Define a Query
query = "select <colunas: tabela.coluna>
        from <tabelas>
        where <condição 01> and <condição 02>.... and <condição n>
        order by <tabela.coluna>";

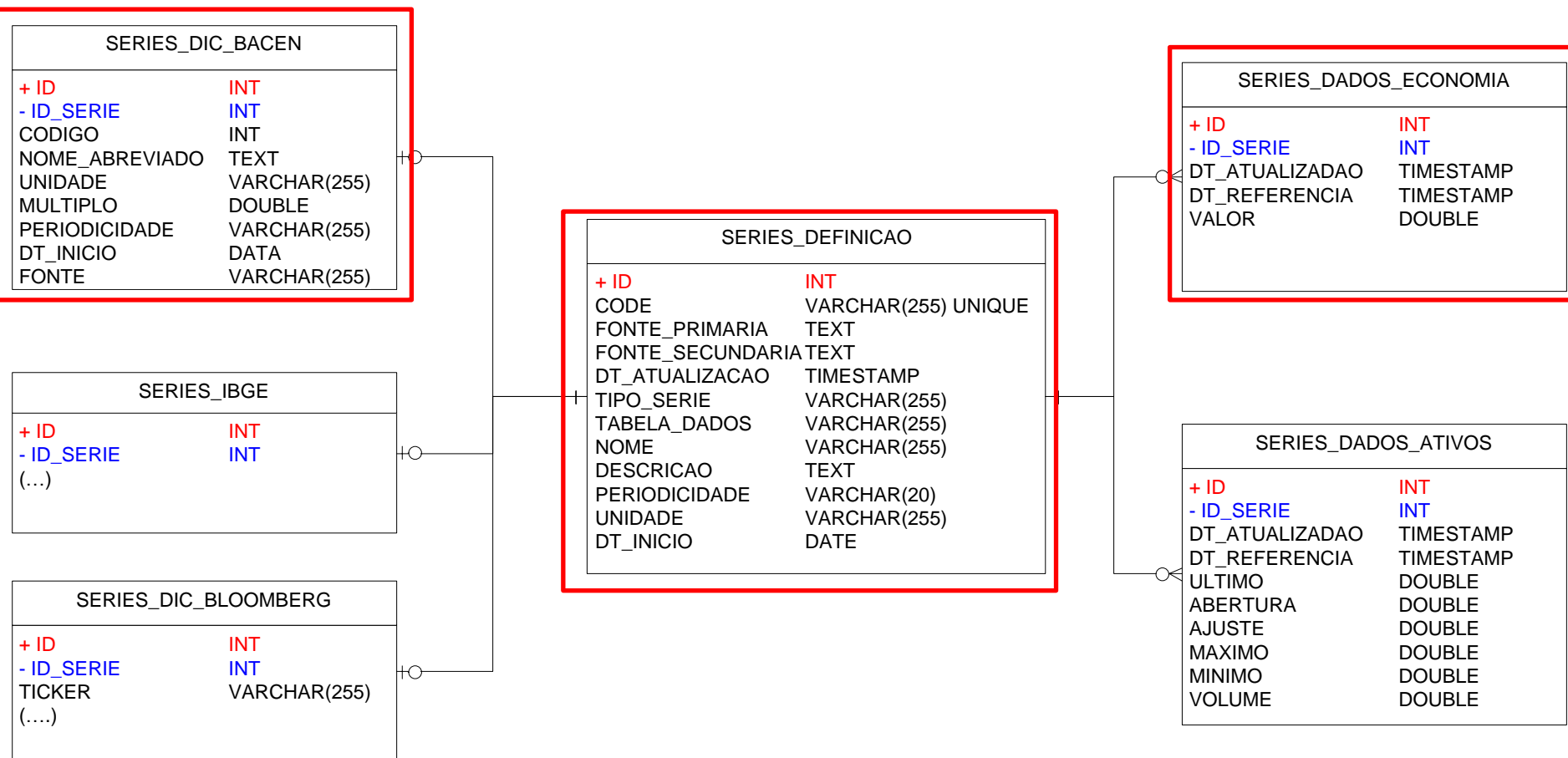
## Executa a query
result = sqlQuery(conn,query,stringsAsFactors=FALSE);

## Fecha a conexão com o banco de dados
odbcClose(conn);

### programa segue....
```

# Consulta a Bancos de Dados

Considere o bando de dados:



Base de dados Access 2007: MQA-PassThrough-Database.accdb

# Consultas a Bancos de Dados

Utilizando o *package* RODBC

1. Escrever um script que seleciona da tabela: `SERIES_DEFINICAO`

Os informações referentes ao índice `IPCA`.

2. Escrever um script que combina as tabelas: `SERIES_DADOS_ECONOMIA`, `SERIES_DEFINICAO` para obter os dados na tabela referentes ao índice `IPCA` limitando o período de Janeiro de 2006 a Dezembro de 2014.

3. Escrever um script que combina as tabelas: `SERIES_DADOS_ECONOMIA`, `SERIES_DEFINICAO` para obter os dados do índice `IPCA`, `IGPM`, `IPDI` limitando o período de Janeiro de 2006 a Dezembro de 2014.

Utilizar da função `reshape(.)` para combinar e tabular a resposta de forma de a primeira coluna seja “Data” e as demais sejam os Código procurados.

# Exercício: Consultas a Bancos de Dados

## Solução Exercício 03:

```
## - Carrega os dados
databasefile = "../database/MQA-PassThrough-Database.accdb"

## Abre a conexão com o banco de dados
conn = odbcConnectAccess2007(databasefile);

## Define a Query
query = "select CODE, DT_REFERENCIA, VALOR
        from SERIES_DEFINICAO, SERIES_DADOS_ECONOMIA
        where SERIES_DEFINICAO.ID = SERIES_DADOS_ECONOMIA.ID_SERIE and
        SERIES_DEFINICAO.CODE IN ('IPCA', 'PTAX', 'IGPM', 'IGPDI') and
        DT_REFERENCIA > #2005-01-01# and DT_REFERENCIA < #2015-01-01#";

## Executa a query
result = sqlQuery(conn, query, stringsAsFactors=FALSE);

## Fecha a conexão com o banco de dados
odbcClose(conn);

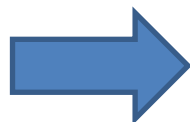
## Coloca os dados em uma Data.frame "wide"
data = reshape(result, idvar="DT_REFERENCIA", timevar="CODE", direction="wide")
colnames(data) = gsub(".*\\.(.+)", "\\1", colnames(data))
```

Coluna comum a todas as séries

Coluna com a identificação das séries

> head(result, 15)

	CODE	DT_REFERENCIA	VALOR
1	IGPM	2005-02-01	0.0030
2	IGPM	2005-03-01	0.0085
3	IGPM	2005-04-01	0.0086
4	IGPM	2005-05-01	-0.0022
5	IGPM	2005-06-01	-0.0044
6	IGPM	2005-07-01	-0.0034
7	IGPM	2005-08-01	-0.0065
8	IGPM	2005-09-01	-0.0053
9	IGPM	2005-10-01	0.0060
10	IGPM	2005-11-01	0.0040
11	IGPM	2005-12-01	-0.0001
12	IGPM	2006-01-01	0.0092
13	IGPM	2006-02-01	0.0001
14	IGPM	2006-03-01	-0.0023
15	IGPM	2006-04-01	-0.0042



> head(data, 15)

	DT_REFERENCIA	IGPM	IGPDI
1	2005-02-01	0.0030	0.0040
2	2005-03-01	0.0085	0.0099
3	2005-04-01	0.0086	0.0051
4	2005-05-01	-0.0022	-0.0025
5	2005-06-01	-0.0044	-0.0045
6	2005-07-01	-0.0034	-0.0040
7	2005-08-01	-0.0065	-0.0079
8	2005-09-01	-0.0053	-0.0013
9	2005-10-01	0.0060	0.0063
10	2005-11-01	0.0040	0.0033
11	2005-12-01	-0.0001	0.0007
12	2006-01-01	0.0092	0.0072
13	2006-02-01	0.0001	-0.0006
14	2006-03-01	-0.0023	-0.0045
15	2006-04-01	-0.0042	0.0002

# Exercício: Estimação Modelo ARIMA

Utilizando o *package* RODBC, vars, urca e forecast:

1. A partir da base de dados um script que combina as tabelas: SERIES\_DADOS\_ECONOMIA, SERIES\_DEFINICAO e monta um data.frame cuja primeira coluna será a Data e as demais os valores dos índices IPCA, IGPM, IPDI, PTAX para o período de Janeiro de 2006 a Dezembro de 2014.

## 2. Análise da série de IPCA:

- Fazer o gráfico de linhas Data vs IPCA
- Utilizar a função `acf()` para fazer o gráfico da auto-correlação
- Utilizar a função `pacf()` para fazer o gráfico da auto-correlação parcial
- Utilizar a função `auto.arima()` para estimar o modelo ARIMA(p,d,q) automaticamente
- Repetir o procedimento `acf()` e `pacf()` para a primeira diferença



# Exercício: *Pass-Through* Câmbio - Inflação

Utilizando o *package* RODBC, vars, urca e forecast:

1. A partir da base de dados um script que combina as tabelas: SERIES\_DADOS\_ECONOMIA, SERIES\_DEFINICAO e monta um data.frame cuja primeira coluna será a Data e as demais os valores dos índices IPCA, IGPM, IPDI, PTAX para o período de Janeiro de 2006 a Dezembro de 2014.

2. Transformação dos dados:

- 1) Série IPCA100: série de IPCA para uma série de IPCA acumulado a partir da base 1 para o início da série

- 2) Série: DLN\_IPCA100: série da primeira diferença o logarítimo natural da série IPCA100

- 3) Série: DLN\_PTAX: série da primeira diferença o logarítimo natural da série PTAX

# Exercício: *Pass-Through* Câmbio - Inflação

(...continuação)

3. Para as séries IPCA, IPCA100 e DLN\_IPCA100 Fazer o teste de ADF para verificação de raiz unitária : `ur.df()`

4. Criar uma `data.frame` auxiliar, `vardata`, com as séries : DLN\_IPCA100 e DLN\_PTAX que será utilizado para estimar os modelos VAR

- Utilizar os comandos:

`VARselect(.)` para selecionar a ordem do modelo VAR

`VAR(.)` para construir um modelo VAR

5. Aplicar os Testes diagnósticos:

- Correlação serial `serial.test()`
- Normalidade `normality.test`

5. Utilizar a função `irf(.)` para estimar o impulso-resposta da PTAX sobre o IPCA

# Exercício: Inserção no Bancos de Dados

Utilizando o *package* RODBC:

1. Escrever uma função que recebe o nome de um arquivo txt com os dados e **força a inserção** dos novos dados na base de dados SQL.
2. Escrever uma função que recebe o nome de um arquivo txt com os dados e **atualiza ou insere** o conteúdo do na base de dados.