UNIVERSIDADE DE SÃO PAULO
FACULDADE DE SAÚDE PÚBLICA

VI Programa de Verão - 2004

Stata - *Básico*

Denise Pimentel Bergamaschi
Milena Baptista Bueno
José Maria Pacheco de Souza

Apresentação

Métodos estatísticos para análise de dados são utilizados por pesquisadores de diversas áreas: economia, sociologia, ciências políticas, marketing, epidemiologia, nutrição, saúde pública. Para o processo de análise dos dados, os pesquisadores necessitam de pacotes que sejam de fácil manejo (amigáveis) e tenham uma ampla gama de técnicas estatísticas.

É o caso do *software* **Stata,** que oferece uma variedade de técnicas estatísticas das mais elementares às mais sofisticadas, tem uma sintaxe simples e é usado por meio de linha de comandos de fácil execução. Foi desenvolvido no Texas (EUA), em 1984, e já é distribuído para 132 países. Periodicamente, o grupo que desenvolve este programa *(StataCorp)* disponibiliza atualizações via internet e tem lançado novas versões a cada três anos, em média. O *StataCorp* também mantém a publicação de um periódico *(Stata Journal)* e uma lista de discussão virtual.

No ano de 2000, dois professores do Departamento de Epidemiologia da Faculdade de Saúde Pública/USP (DPB e JMPS) ministraram um curso básico do Stata, no Programa de Verão dessa Faculdade. Desde então, observou-se grande interesse dos docentes, pesquisadores e alunos de pós-graduação, o que estimulou a repetição do curso. Nesse período, o material utilizado sofreu modificações, inclusive a atualização para a versão 7.0**.** Esse manual foi o utilizado no último Programa de Verão, em janeiro de 2004; os arquivos usados nos exemplos e exercícios podem ser baixados acessando a página

http://www.fsp.usp.br/~jmpsouza/statabasico.

Os autores.

São Paulo, 5 de julho de 2004

Índice

1 - Iniciando o trabalho no Stata

Stata [Estata ou Esteita] - *Stata Corporation*

- *Intercooled Stata*
- *Versão resumida - Short Stata*
- Versão simplificada *StataQuest*

Existem versões do programa para 3 sistemas: *Windows*, *Unix* e *Macintosh*. Atualmente está na versão 8.

Este curso: Intercooled Stata versão 7 para sistema *Windows*.

O Stata é descrito em um manual com 5 volumes e em *Hamilton* (1998).

Cada comando está associado a um arquivo-**help** que pode ser acessado durante a utilização do programa.

Informações sobre o Stata, bem como atualizações, realização de cursos via *Internet* e lista das dúvidas mais freqüentes podem ser obtidas no *site*: http://www.stata.com.

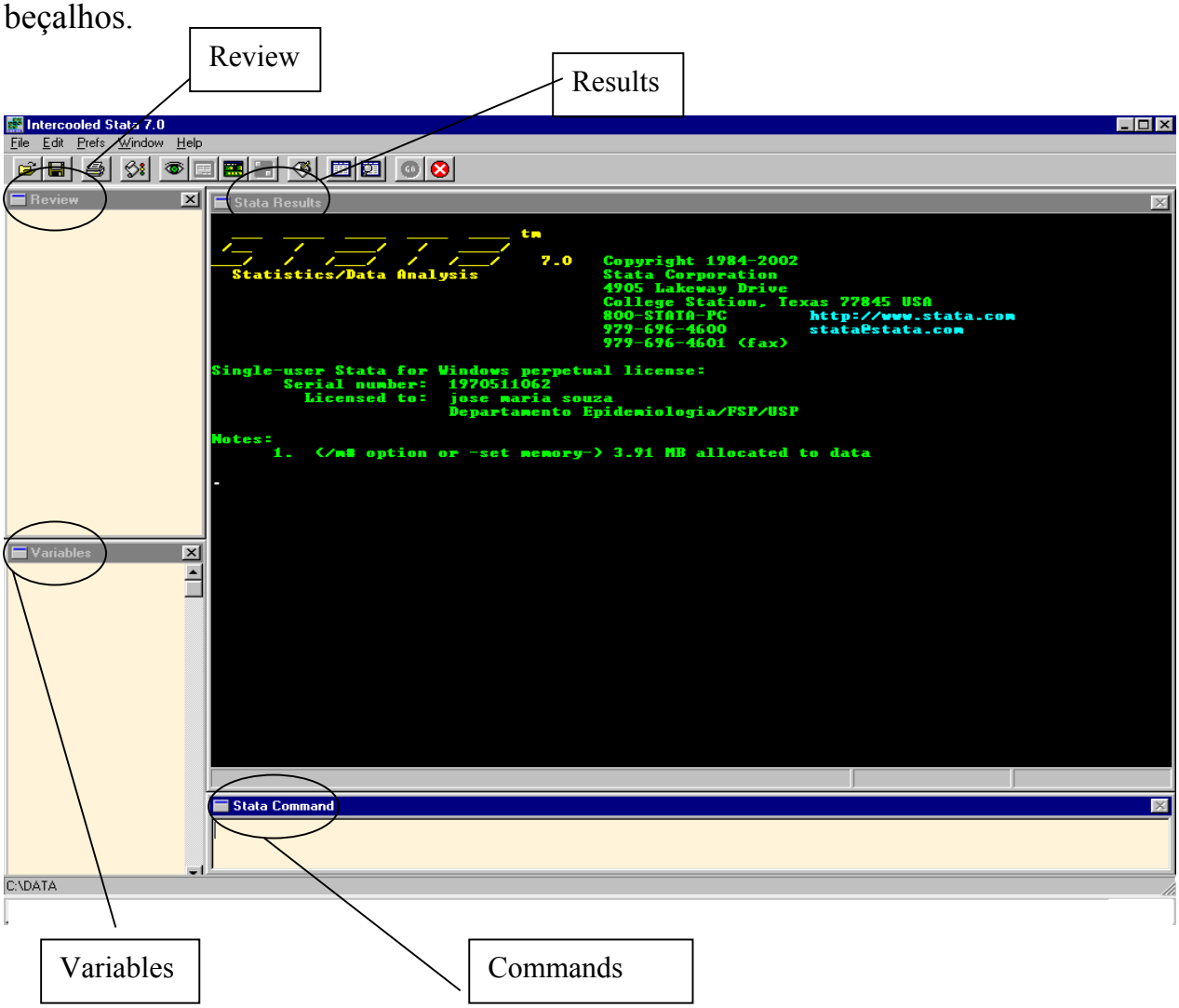
O Stata possui lista de discussão sobre dúvidas
endereço: statalist@hsphsun2.havard.edu

1.1 - Iniciando o Stata

Abrir o programa

- diretamente pelo ícone na tela de abertura do *Windows*, ou
- seguindo o caminho **Iniciar**, **Programas**, **Stata, Intercooled Stata 7**

Quando o programa é aberto, abre-se uma tela contendo janelas menores, com cabeçalhos.



A finalidade de cada janela é apresentada a seguir:

Título	Finalidade
Review	Armazenamento dos comandos já utilizados. O comando pode ser reutilizado e corrigido utilizando-se o mouse ou as teclas **PgUp** (*page up*) e **PgDn** (*page down*)
Variables	Apresentação das variáveis contidas no banco de dados.
Stata Results (fundo preto)	Apresentação dos resultados obtidos com a execução dos comandos
Stata Command	Digitação dos comandos a serem executados. Digitar quando o *prompt* estiver ativo. Executar pressionando a tecla **Enter.** Os comandos devem ser digitados em letra minúscula e as variáveis como foram criadas.

1.2- Tipos de arquivos

O Stata trabalha com 4 tipos de arquivos:

Tipo de arquivo	Extensão
Arquivo que contém os dados	.dta
Arquivo que guarda os comandos e resultados obtidos durante a sessão de trabalho	.log; .smcl
Arquivo que contém comandos	.do
Arquivo que contém sub-rotinas	.ado

Logo que for iniciado o trabalho no Stata, é aconselhável abrir um arquivo **log,** que armazenará todos os comandos e seus resultados (com exceção de gráficos).

Para abrir um arquivo log:

Clicar sobre o quarto ícone (pergaminho) e salvar como tipo log

Ou

Digitar:

- log using <diretório:\nome do arquivo>

O arquivo **log** é um arquivo de tipo somente texto e não permite alteração. Caso seja de interesse, pode-se abri-lo em um editor de textos, por exemplo no *Word for Windows* e salvá-lo com extensão .doc para ser manipulado segundo a necessidade.

A extensão .smcl não será abordada neste curso.

1.3 – Sintaxe dos comandos

O Stata é um programa de comandos. Os comandos seguem a forma:

[by varlist:] command [varlist] [weight] [if exp] [in range] [using filename] [,options]

onde

[by varlist:] instrui o Stata para repetir o comando para cada combinação de valores nas variáveis listadas em *varlist*;

command é o nome do comando, ex: **list**

[varlist] é a lista de variáveis para as quais o comando é executado

[weight] permite que pesos sejam associados às observações

[if exp] restringe o comando a um subconjunto de observações que satisfazem a expressão lógica definida em *exp*

[in range] restringe o comando àquelas observações cujos índices pertencem a um determinado subconjunto

[using filename] especifica o arquivo que deve ser utilizado

[,options] são específicas de cada comando.

Ex: usando um banco de dados contendo as variáveis **x** e **y**
o comando para listá-las é: **list x y**
pode ser definida uma condição (if)**: list x y if x>y**

O programa diferencia entre letra maiúscula e minúscula. Todos os comandos devem ser em letra minúscula e as variáveis de acordo como foram criadas.

A utilização do **Help** é fortemente recomendada; clicando-se em **Help** no menu principal, pode-se pesquisar qualquer comando utilizando-se a opção **Contents (todo o manual)**, **Search (palavras chaves)** ou **Stata command (comando)**.

O programa funciona com o mínimo de memória e tamanho de matriz para tornar o programa mais ágil. Quando, ao abrir um banco de dados, surgir o aviso de memória insuficiente, a memória pode ser expandida pelo comando:

- **set memory 32m**

Quando, ao abrir um banco de dados, surgir o aviso que o tamanho da matriz é insuficiente, a matriz pode ser aumentada pelo comando:

- **set matsize 400**

Os comandos que iniciam com *set* alteram o *default* de configurações do programa e devem ser realizados sem arquivo aberto.

1.4- Arquivos de dados em formato <u>não dta</u>

O Stata lê arquivos ASCII (extensão .raw e .dat) e arquivos texto (.txt). Existe a possibilidade do banco de dados ser construído em uma planilha do Excel e ser transferido para o Stata. No Excel, na primeira linha do banco pode-se digitar o nome das variáveis. Os valores faltantes devem ser substituídos por valores numéricos (-99, p.ex.). Salvar como texto (separado por tabulação), ou seja, extensão .txt.

ATENÇÃO: Variáveis numéricas com casas decimais ⟶ Não digitar vírgula para separar casas decimais e sim ponto. Ex. 9.125

A leitura no Stata é com o comando **insheet**.

- **insheet using c:\<subdiretorio>\<arquivo>.txt**

Exercício: Transferir os dados digitados no arquivo c:\cursosta\glicose.xls para o Stata.

Pode-se utilizar o *Transfer* ou outro pacote que realize conversão de bancos de dados.

1.5 – Criando banco de dados no Stata

Entrada de dados diretamente no Stata, pelo teclado

- **input [varlist]**

Criar um banco de dados com nome **banco1** que contenha as variáveis id, nome, tratamen, pesoinic e sexo; para 10 pacientes, com dados apresentados a seguir.

id	nome	tratamen	pesoinic*	sexo
1	A Silva	0	98.405	1
2	G Soares	1	75.505	2
3	V Gomes	1	93.699	2
4	M Costa	0	80.22	1
5	A Sena	0	70.0	2
6	V Jardim	0	-99	1
7	B Rolim	1	89.65	2
8	N Rego	1	-99	1
9	F Dias	0	-99	2
10	H Bastos	1	50.659	2

*** -99 = sem informação**

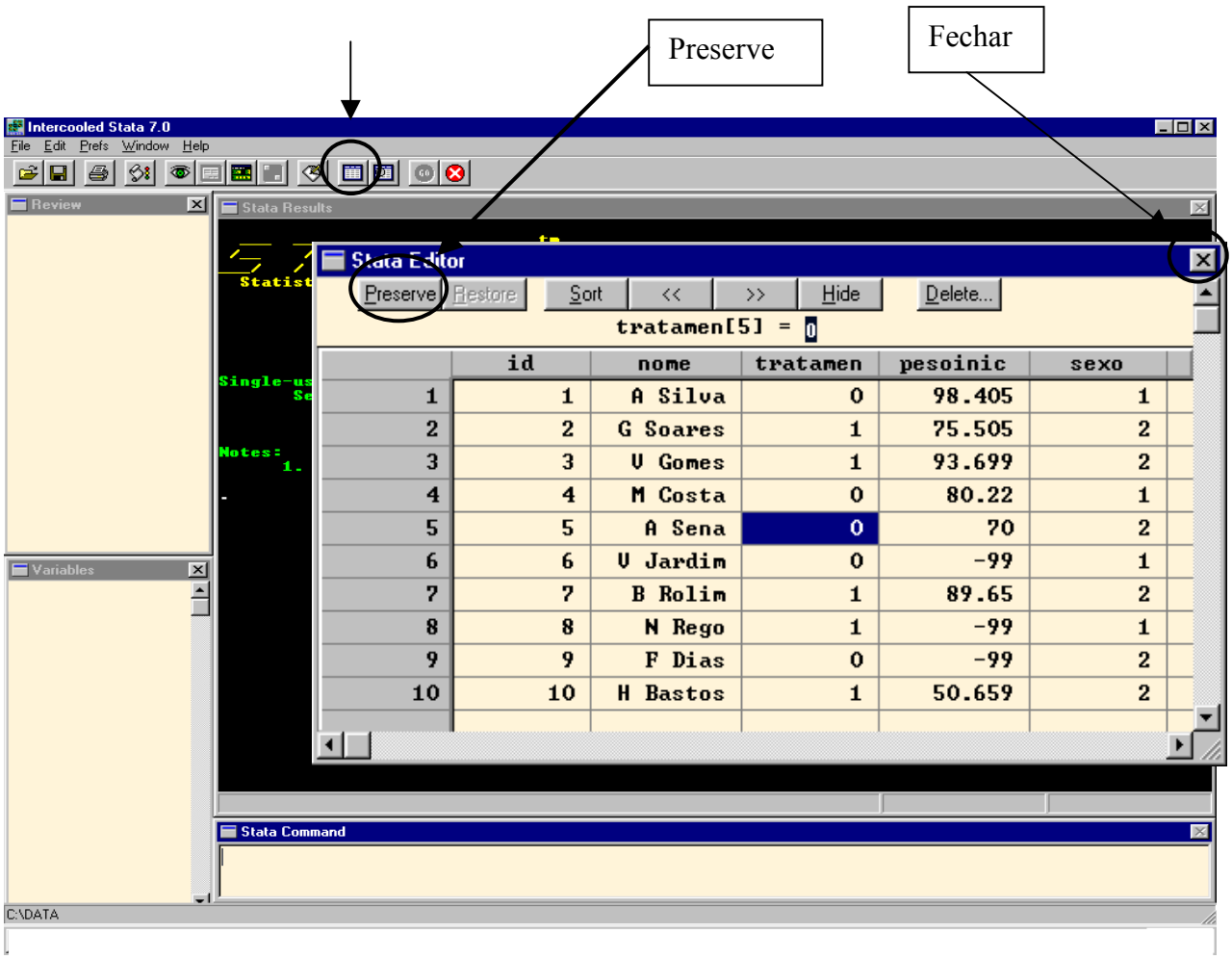
- **input id str10 nome tratamen pesoinic sexo**
- **1 "A Silva" 0 98.4 1**
- **end**

str10: indica que a variável a seguir é do tipo *string* com 10 caracteres. Se não houver esta informação a variável criada será numérica.

"A Silva" precisou usar aspas porque tem espaço entre as palavras. Se fosse Asilva, não precisaria.

É recomendável criar variáveis com nome em letra minúscula e sem acentos ou cedilhas a fim de facilitar a manipulação do banco de dados.

Abrir modo de edição clicando sobre o ícone **Data editor** (10º ícone do menu com desenho de uma planilha) e digitar os dados dos demais registros. Usar **Tab** para entrada horizontal e **Enter** para entrada vertical. Para digitar letras no **data editor** não é mais necessário utilizar aspas. Quando terminar, pressionar **Preserve** seguido de **Close** no menu do Stata editor (ou pressionar a cruz do lado direito da tela).



	id	nome	tratamen	pesoinic	sexo
1	1	A Silva	0	98.405	1
2	2	G Soares	1	75.505	2
3	3	V Gomes	1	93.699	2
4	4	M Costa	0	80.22	1
5	5	A Sena	0	70	2
6	6	V Jardim	0	-99	1
7	7	B Rolim	1	89.65	2
8	8	N Rego	1	-99	1
9	9	F Dias	0	-99	2
10	10	H Bastos	1	50.659	2

Pode-se utilizar o editor desde o início. Os nomes das variáveis são inseridos clicando duas vezes na primeira célula da coluna (cinza) após a digitação de algum valor na coluna.

ATENÇÃO: o banco de dados ainda não está salvo.

1.6 –Salvamento e leitura de banco de dados

Para salvar um arquivo

O arquivo deve ser salvo utilizando a caixa de diálogo, na seqüência:

File, **Save As**, Sub-diretório - **cursosta**, nome do arquivo: **banco1**

Ou

Digitando na linha de comando:

- **save c:\cursosta\banco1.dta**

Se o arquivo já existir e deseja-se salvar as alterações, digitar:

- **save c:\cursosta\banco1.dta, replace** *ou* **save, replace**

Para fechar um arquivo sem salvar e sem abrir outro arquivo:

- **clear**

Para abrir um arquivo

Pressionar o *mouse* sobre **File** seguido de **Open**. Seleciona-se o sub-diretório que contém o arquivo ***.dta***, marca-se o arquivo e seleciona-se **Open**.

Ou

Digitar na linha de comando:

- **use c:\cursosta\banco1.dta, clear**

Para ler um arquivo no Stata, é necessário limpar os dados utilizados anteriormente e que podem estar ainda na memória de execução do programa, por isso a necessidade da opção *clear* no comando.

1.7 – Manipulação de variáveis

Há dois tipos de variáveis no Stata: *string* (caracteres, letras) e numérica. Estas variáveis são armazenadas de formas diferentes que requerem tamanhos diferentes nos registros de memória: *byte*, *int*, *long* e *float* para variáveis numéricas e *str1* até *str80* para variáveis *string* de tamanhos diferentes. Além disto, cada variável pode ter um nome associado a ela (rótulo, *label*) e tem um formato de apresentação

O rótulo da variável pode ser definido com o comando:

- **label var pesoinic "peso inicial"**

Para descrever o formato e os rótulos das variáveis digitar:

- **describe**

O formato de uma variável pode ser modificado com o comando *format*:

- **format pesoinic %7.2f**

Modifica o tamanho da variável numérica *pesoinic*: 7 espaços antes da virgula e 2 casas decimais após a virgula em um formato fixo (f).

- **format nome %15s**

Modifica o tamanho da variável string *nome*: 15 espaços ao invés de 10.

O nome da variável *x* pode ser mudado para *y* usando o comando

- **rename x y**
- **rename nome paciente**

Variáveis numéricas

Valores faltantes (*missing*) são representados por pontos e são interpretados como valores muito grandes.

Substitui todos os valores de *x* iguais a -99, para pontos (.)

- **mvdecode x, mv(-99)**
- **mvdecode _all, mv(-99)**

O código de valores *missing (.)* pode ser convertido em valores, como -99:

- **mvencode x, mv(-99)**
- **mvencode _all, mv(-99)**

Definição de rótulos para categorias de variáveis:

- **label define titulos 1 "casado" 2 "divorciado" 3 "viuvo" 4 "solteiro"**
- **label values marital titulos**

Ex:

- **label define s 1 "masculino" 2 "feminino"**
- **label values sexo s**
- **tab sexo**
- **tab sexo, nolabel**

OBS: Quando o label for igual para várias variáveis é possível direcionar um único label para todas estas.

Recodificação de variáveis:

- **recode marital 1 2 =2 4=3** ou
- **recode marital 1/2=2 4=3**

Ex:

- **recode sexo 1=0 2=1**

Variáveis *string*

Variáveis *string* são utilizadas para variáveis com categorias não numéricas, sob a forma de palavras, ou, genericamente, um conjunto de caracteres, com ou sem sentido de palavra. São representadas por %Xs (X = nº de caracteres)

Variáveis data

O Stata lê variáveis data como tempo decorrido (*elapsed dates)* ou **%d**, que é o número de dias contados a partir de 01 de janeiro de 1960. Assim,

0 corresponde a	01jan1960
1 corresponde a	02jan1960
.	.
.	.
.	.
15000 corresponde a	25jan2001

O Stata possui funções para converter datas em **%d**, para imprimir **%d** em formatos compreensíveis.

Variáveis datas devem ser definidas como variáveis *string* e depois convertidas para **%d**.

- **input str10 datanasc**

Digitar:

datanasc
“12/04/1955”
“14/03/1960”
“15/07/1954”
“12/06/1969”
“5/10/1970”
“20/03/1967”
“21/09/1971”
“31/03/1958”
“25/08/1972”
“26/01/1970"

- **end**
- **gen dianiver=date(datanasc,"dmy")**
- **list datanasc dianiver**
- **desc**
- **format dianiver %d**
- **list datanasc dianiver**
- **gen idade=(date("28/01/2004", "dmy")-dianiver)/365.25**
- **list datanasc dianiver idade**

2 - Manipulação de dados

2.1 - Expressões

Existem expressões lógicas, *string* e algébricas, no Stata.

Expressões lógicas atribuem 1 (verdadeiro) ou 0 (falso) e utiliza os operadores:

Operador	Significado
<	menor que
<=	menor ou igual a
>	maior que
>=	maior ou igual a
= =	igual a
~= !=	diferente de
~	não
&	e
|	ou

Ex: **if (y~=2 & z>x) | x= =1**

Significa: se (**y** for diferente de 2 e **z** maior do que **x**) ou **x** for igual a 1

Expressões algébricas utilizam os operadores:

Operador	Significado
+ -	soma, subtração
* /	multiplicação, divisão
^	elevado à potência
sqrt( )	função raiz quadrada
exp( )	função exponencial
log( )	função logarítmica (base 10)
ln( )	função logarítmica (base *e*) - logaritmo natural

Abrir o arquivo c:\cursosta\sistolica.dta

- **use c:\cursosta\sistolica.dta**

Abrir um arquivo c:\cursosta\sistolica.log para armazenar os resultados

- **log using c:\cursosta\sistolica.log**

Os dados que serão utilizados nesta sessão constituem uma amostra de 58 pacientes hipertensos, do sexo feminino que foram avaliados por 6 meses. As variáveis estudadas foram:

- **droga**: tipo de medicamento utilizado no período (1=nenhum; 2=tipo A; 3=Tipo B; 4=Tipo C)
- **sistolica**: incremento da pressão sistólica
- **idade**: idade em anos
- **salario**: renda do paciente (R$)
- **familia**: número da família (tem pacientes da mesma família).
- **pesoin:** peso inicial (kg) do paciente
- **pesointer:** peso (kg) do paciente após 3 meses de tratamento
- **pesof:** peso (kg) após 6 meses de tratamento

2.2 - Observações índice e conjunto de valores

Observações índice

Cada observação está associada a um índice. Por exemplo, o terceiro valor da variável *x* pode ser especificado como **x[3]**. O macro **_n** assume um valor para cada observação e **_N** é igual ao número total de observações. Pode-se referir à penúltima observação da variável *x* escrevendo-se **x[_N-1]**.

Uma variável indexada deve ficar do lado direito de uma asserção. Por exemplo, para substituir a terceira observação da variável *x* pelo valor 2 escreve-se:

- **replace droga=2 if _n==3**

Conjunto de valores

Um conjunto de valores pode ser especificado utilizando-se **if** ou utilizando **in range** que possui a sintaxe **f/l (f para *first* e l {letra ele} para *last*)**. Exemplos:

- **list sistolica in –10/l (os últimos 10 registros)**
- **list sistolica in 10/l (do décimo registro ao ultimo)**

Para repetir comandos para variáveis ou categorias de variáveis, utilizar **by varlist;** os dados precisam estar ordenados antes disto, o que é feito utilizando o comando **sort**.

- **sort droga**
- **by droga: list sistolica**

2.3 - Gerando variáveis

O comando **generate** cria uma nova variável igualando à uma expressão que é construída para cada observação.

- **generate <nome var>=<expressão>**

Ex:

- **generate id=_n**

Gera um nova variável *id* na qual cada indivíduo terá um número de identificação que será o mesmo que a observação índice.

- **generate porpeso= ((pesof-pesoin)/pesoin)*100**

Gera uma nova variável *porpeso* que será a porcentagem de variação do peso em relação ao peso inicial. Assumirá valor faltante se **pesoin** ou **pesof** for valor faltante ou será igual ao percentual de diminuição de peso em relação ao peso inicial.

- **generate incrsistolica=0 if sistolica<0**

Cria a variável *incrsistolica* que categorizará os indivíduos entre os que tiveram aumento ou diminuição da pressão sistólica durante o período de observação. O valor 0 indicará diminuição da pressão. O restante dos indivíduos serão codificados como valores faltantes. .

O comando **replace** funciona como o comando **generate,** com a diferença que permite que uma variável já existente seja alterada.

- **replace <nova var>=<nova expressão> <condição>**

Ex:

- **replace incrsistolica=1 if sistolica>=0**

Modifica os valores faltantes para 1 se *sistolica* maior ou igual a 0.

- **replace porpeso=-99 if porpeso==.**

Modifica o valor faltante da variável *porpeso* por -99.

Gerando variáveis indicadoras (dummy)*:*

A variável droga é categorizada em 1, 2, 3 e 4. O comando:

- **tab droga, gen(droga)**

Gera 4 variáveis *dummy*: droga1, droga2, droga3 e droga4 de tal forma que droga1 terá valores iguais a 1 quando a droga utilizada for a 1 e 0 se a droga utilizada for 2, 3 ou 4. A variável droga2 terá valores iguais a 1 quando a droga utilizada for a 2 e 0 se a droga utilizada for 1, 3 ou 4. E assim será para as variáveis droga3 e droga4.

Variáveis dummy terão aplicação, por exemplo, na construção de gráficos de pizza e análise multivariada..

Comando **egen:**

O comando **egen** pode ser função de muitas variáveis simultaneamente.

- **egen media=rmean(pesoin-pesof)**

Cria uma nova variável e calcula a média de peso para cada indivíduo utilizado o peso inicial e peso final. Os valores faltantes são ignorados.

rmean trabalha nas linhas**.**

- **egen famsal=mean(salario),by(familia)**

Cria uma nova variável e calcula a média da variável **salario** para o conjunto de valores iguais de familia**.**

mean trabalha na coluna da variável.

Uma variável existente pode ser retirada do banco de dados com o comando **drop.**

- **drop salario**

Pode-se utilizar, também, o comando **keep varlist**, onde **varlist** é a lista de variáveis que devem permanecer no banco de dados.

SALVAR O ARQUIVO - pelo menu ou pelo comando:

- **save, replace**

FECHAR O BANCO DE DADOS:

- **clear**

FECHAR O ARQUIVO LOG - pelo pergaminho ou pelo comando:

- **log close**

Abrir o arquivo log no *Word for Windows.*

2.4 - Mudando a forma de apresentação dos dados

Supor a situação na qual, para um mesmo indivíduo, são obtidas duas ou mais informações, apresentadas no banco de dados c:\cursosta\calorias.dta.

Os dados estão apresentados como segue, em formato **wide**.

- **use c:\cursosta\calorias.dta**
- **list**

	id	cal1	cal2	sexo
1.	1	2300	2500	1
2.	2	2400	3200	1
3.	3	2400	3600	1
4.	4	3200	3500	2
5.	5	3000	3200	2
6.	6	3000	3500	2
7.	7	2564	3589	1
8.	8	2600	2785	1
.	.	.	.	.
.	.	.	.	.
19.	19	3800	3500	1
20.	20	2980	2851	2

A forma de apresentação dos dados pode ser mudada para o formato **long**, utilizando o comando

- **reshape long cal, i(id) j(consulta)**
- **list**

	id	consulta	cal	sexo
1.	1	1	2300	1
2.	1	2	2500	1
3.	2	1	2400	1
4.	2	2	3200	1
5.	3	1	2400	1
6.	3	2	3600	1
7.	4	1	3200	2
8.	4	2	3500	2
9.	5	1	3000	2
10.	5	2	3200	2
11.	6	1	3000	2
12.	6	2	3500	2
.	.	.	.	.
.	.	.	.	.
39.	20	1	2980	2
40.	20	2	2851	2

Para reverter ao formato anterior (**wide**)

- **reshape wide cal, i(id) j(consulta)**
- **list**

2.5- Unir bancos de dados

O arquivo que está aberto (calorias.dta) é denominado mestre.

Objetivo 1: Acoplar os dados de um segundo banco ao final do banco mestre, como em continuação deste. Não precisa ter necessariamente as mesmas variáveis.

- **append using <arquivo>**

Ex:

- **append using c:\cursosta\calorias2.dta**
- **save c:\cursosta\calorias12.dta**

id	cal1	cal2	sexo
1	2300	2500	1
2	2400	3200	1
3	2400	3600	1
4	3200	3500	2
5	.	.	.
20	.	.	.

Banco Mestre

↓

id	cal1	cal2	sexo	idade
21	2560	2001	1	45
22	2330	2064	1	42
23	2648	2542	1	36
24	2900	2981	2	35
25	.	.	.	.
26	.	.	.	.

Banco 2

Objetivo 2: unir lado a lado dois bancos de dados que contenham informações correspondentes à mesma unidade de observação (indivíduo, família, animal, etc...). É necessário que os bancos tenham uma variável de identificação (com a mesma sintaxe) e que esteja ordenado por esta variável.

- **merge <variável de identificação> using <arquivo>**

Ex:

- **sort id**
- **save, replace**

Abrir o segundo banco

- **use c:\cursosta\sintomas**
- **sort id**
- **save, replace**
- **use c:\cursosta\calorias12.dta**
- **merge id using c:\cursosta\sintomas**

id	cal1	cal2	sexo	Idade
1	2300	2500	1	.
2	2400	3200	1	.
3	2400	3600	1	.
4	3200	3500	2	.
5	.	.	.	.
21	.	.	.	45

→

id	enjoo	fome	diarreia	febre
1	2	1	1	2
2	2	2	2	2
3	1	2	2	2
4	1	2	2	1
.	.	.	.	.
.	.	.	.	.

Banco Mestre ***Banco 2***

O comando *merge* gera uma variável *_merge* com os códigos:

1- dados faltantes no banco 2

2- dados faltantes no banco mestre

3- união de dados realizada com sucesso

Salvar o banco de dados com o nome: inteiro.dta

3. Descrição de dados

* Abrir o banco de dados sistolica.dta

3.1 - Gráficos

A sintaxe básica para a elaboração de gráficos é:

- **graph varlist, options**

Em **options** deve-se especificar o tipo de gráfico desejado.

Os gráficos não aparecem no arquivo log. Deve-se abrir um arquivo .doc previamente; obtido o gráfico, clicar em **copy graph** na barra do Stata e depois **colar** no doc.

Boxplot

- **graph idade, box**

Produz um boxplot da variável idade

- **graph pesoin pesof, box ylabel**

Cria dois boxplots, um para pesoin e outro para pesof, em um mesmo conjunto de eixos ortogonais. A opção *ylabel* faz com que o nome da variável apareça no eixo y.

- **by droga: graph idade, box**

Fornece um boxplot para cada categoria de droga, em dois conjuntos de eixos ortogonais independentes. Antes deste comando é preciso ordenar os dados pela *droga*. Os comandos que utilizam a expressão *by* geralmente solicitam dados ordenados. O comando é: **sort droga**

- **graph idade, by(droga) box**

Cria boxplots, um para cada categoria de droga, em um mesmo par de eixos

Diagrama de dispersão

- **graph idade pesoin, xlabel ylabel t1("diagrama de dispersão")**

Fornece um diagrama de dispersão de idade e pesoin

As opções **xlabel** e **ylabel** fazem com que os eixos *X* e *Y* sejam rotulados utilizando valores redondos das varíaveis idade e pesoin (sem estas opções serão apresentados somente os valores mínimo e máximo).

A opção **t1("diagrama de dispersão")** faz com que seja apresentado um título principal no topo do gráfico. **b1( )**, **l1( )** e **r1( )** produzem títulos principais na base, na esquerda e direita. **t2( )**, **b2( )**, **l2( )** e **r2( )** produzem títulos secundários em cada um dos lados.

Histograma

- **graph idade**

Desenha um histograma da variável idade.

- **graph idade, bin(10)**

Desenha um histograma da variável **idade** em 10 intervalos de classe. O número de intervalos pode variar, de acordo com os dados.

- **graph idade, bin(10) norm**

Desenha um histograma da variável idade com 10 intervalos de classe e sobrepõe uma curva normal com a média e o desvio padrão observados.

- **graph idade, bin(10) norm(média desviopadrão)**

Desenha um histograma da variável idade para cada tipo de tratamento com 10 intervalos de classe e sobrepõe uma curva normal com média e desvio padrão definidos.

- **graph idade, bin(10) xlabel (20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75) ylabel t1("distribuição da idade") freq**

Desenha um histograma da variável x com 10 intervalos de classe, apresenta os rótulos dos eixos, o título (no topo do gráfico) e no eixo y, o número de indivíduos ao invés do percentual.

- **graph idade, bin(10) xlabel ylabel by(droga)**

desenha um histograma da variável idade, com 10 intervalos de classe, para cada categoria da variável droga.

Pizza

- **graph droga1 droga2 droga3 droga4, pie**

A variável droga já deve estar já categorizada em grupos.

Para mudar as características dos gráficos, com o *mouse* selecionar *Prefs* na barra de menu e em seguida *Graph Preferences*.

Para levar um gráfico para o *Word*, com o *mouse* selecionar *Edit* na barra de menu e em seguida *Copy Graph*. No *Word*, colar o gráfico.

3.2 – Tabelas e resumo dos dados

Os dados que serão utilizados nesta sessão constituem uma amostra de 118 pacientes psiquiátricos, do sexo feminino e estão disponíveis em D.J. Hand et al. ***A Handbook of Small Data Sets***. Chapman &Hall, London, 1994. As variáveis estudadas foram:

- **age**: idade em anos
- **iq**: escore de inteligência (-99 = ignorado)
- **anxiety**: ansiedade (1=nenhuma, 2=leve, 3=moderada, 4=severa)
- **depress**: depressão (1=nenhuma, 2=leve, 3=moderada, 4=severa)
- **sleep**: você pode dormir normalmente? (1=sim, 2=não)
- **sex**: você perdeu interesse em sexo? (1=não, 2=sim)
- **life**: você tem pensado recentemente em acabar com sua vida? (1=não, 2=sim)
- **weight**: mudança no peso durante os últimos 6 meses (em libras)

* -99 igual à valor faltante.

Exercício:

Objetivo	Comandos
Abrir o banco de dados	use c:\cursosta\fem.dta
Abrir um arquivo .log (salva os comandos e as tabelas)	log using c:\cursosta\fem.log
Verificar quais são as variáveis que compõem o banco de dados	describe ou desc
Construir uma tabela de freqüências simples de cada variável	tab1 _all Ou pedir uma tabela para cada variável tab age tab anxiety
Remover os valores faltantes	mvdecode _all, mv(-99) Ou removendo os valores faltantes de cada variável: recode sleep –99=.
Recodificar a variável **sleep**, para ficar consistente com o restante dos códigos (1=não e 2=sim)	recode sleep 1=2 2=1
Criar rótulos (*labels*) para as variáveis	label define sn 1 nao 2 sim label values sex sn label val sleep sn label val life sn
Obter a média e intervalo de 95% da variável **weight**	means weight
Obter um resumo da variável **iq**	sum iq ou sum iq,d
Obter um resumo da variável **iq** segundo **life**	sort life by life: sum iq,d
Obter as médias e desvios padrão de **iq** segundo life	table life, contents(mean iq sd iq)

Objetivo	Comandos
Criar um rótulo para a variável **weight**	label variable weight “mudanca de peso nos ultimos 6 meses”
criar rótulo para a variável **life**	label variable life “voce pesnsou em terminar sua life recentemente?”
fazer o gráfico boxplot da variável **weight** segundo **life**	graph weight, box by(life) b1(“voce pensou recentemente em terminar sua vida?”)
fazer o gráfico qq-plot para verificar normalidade da distribuição da variável **weight**	qnorm weight, gap(5) xlab ylab t1(“qq plot para normalidade”) onde, gap(5) é usado para diminuir o espaço entre o eixo vertical e o título do eixo
Desenhar um histograma da variável **weight** em 6 intervalos de classe.	graph weight, bin(6) xlabel(-5, -2.5, 0, 2.5, 5, 7.5) ylabel t1(“distribuição de perda de peso nos ultimos 6 meses”)
Desenhar um histograma da variável **weigh**t em 6 intervalos de classe, segundo a variável **life**	graph weight, bin(6) xlabel(-5, -2.5, 0, 2.5, 5, 7.5) ylabel by(life)
Criar uma variável **ageg** contendo a variável age em intervalos de classes de 5 anos	gen ageg=age recode ageg 25/29=1 30/34=2 35/39=3 40/44=4 45/49=5 label define id 1 “25-29” 2 “30-34” 3 “35-39” 4 “40-44” 5 ”45-49” label val ageg id tab ageg

4. Análise de dados epidemiológicos

Banco de dados: c:\cursosta\fem.dta

Comparação de médias:

Para comparar as variáveis quantitativas entre grupos pode-se utilizar o teste *t de "Student"* que assume que as observações nos dois grupos são independentes; as amostras foram retiradas de populações com distribuição normal, com mesma variância. Um teste alternativo, não paramétrico, que não necessita destas pressuposições, é o teste *U* de *Man-Whitney*. Para mais de dois grupos independentes, utiliza-se a analise de variância (ANOVA) oneway; a análise correspondente na estatística não paramétrica é o teste de *Kruskal-Wallis*.

Coeficiente de correlação:

É possível calcular correlações entre variáveis contínuas. Se se quiser testar se o coeficiente de correlação de *Pearson* é estatisticamente diferente de zero, o Stata apresenta um teste que pressupõe que as variáveis são normais bivariadas. Se esta pressuposição não for feita, pode-se utilizar a correlação de postos de *Spearman*. Se as variáveis forem categóricas é possível utilizar a estatística de *Kendall* como medida de associação.

Associação entre variáveis:

Para as variáveis qualitativas nominais pode-se utilizar o teste qui-quadrado, de *Pearson*.

4.1 – Teste de hipóteses para uma, duas ou três médias e intervalos de confiança

Intervalo de 95% de confiança de média

- **ci weight**

Intervalo de confiança de médias de **weight** segundo **life**

- **sort life**
- **ci weight, by (life)**

Intervalo de 95% de confiança para uma dada amostra, média e desvio padrão

- **cii 100 2 2.5**

Amostra=100; Média observada=2; Desvio padrão populacional=2,5

Teste de diferença de variância entre grupos

- **sdtest weight, by (life)**

Teste de duas média (t de "Student") entre grupos

- **ttest weight, by (life)**

Teste de duas médias pelo método não paramétrica (U de Man-Whitney)

- **ranksum weight, by (life)**

Testar a hipótese de que a média observada é igual a um valor

- **ttest weight=2**

Testa se a média da variável **weight** (1,58) é igual à média populacional (2)

Análise de variância com um fator (ANOVA)

- **oneway weight depress, bonferroni means st**

bonferroni: teste que identifica a categoria significante;
means e st: mostra um quadro resumo contendo a média e o desvio padrão das categorias.

Teste de mais de duas médias pelo método não paramétrica (Kruskal-Wallis)

- **kwallis weight, by (depress)**

*Lembrar de ordenar o banco pela variável depress antes do comando.

4.2 – Teste de hipóteses e intervalo de confiança para proporção

Intervalo de 95% de confiança para proporções

- **tab sleep**
- **cii 112 0.125**

Constrói o intervalo de confiança para a proporção dos 14 pacientes (12,5%) que tem problemas para dormir.

Testar a hipótese de que a proporção observada é igual a um valor

Para este teste é necessário que a variável esteja codificada em 0 e 1, portanto:

- **recode life 1=0 2=1**

- **bitest life=0.5**

  ou

- **bitesti 117 65 0.5**

Testa se a proporção de pessoas que pensaram em se matar é equivalente a 0,5

Associação de variáveis categóricas

Teste qui-quadrado

- **tab life depress, col row chi2**

Teste exato de Fisher

- **tab life sleep, col row exact**

As opções col e row colocam as proporções na tabela

4.3 – Teste de hipóteses para correlação

Calcular a correlação

- **corr weight iq age**

  ou

- **pwcorr weight iq age,sig**

A opção *sig* apresenta a significância estatística.

Calcular a correlação pelo método não paramétrico (Teste de Spearman)

- **spearman weight age**

4.4 - Análise de medidas de efeito

Todos os comandos de estimação seguem a mesma estrutura em sua sintaxe:

[xi:] command depvar [model] [weights],options

A variável resposta é especificada por **depvar** e as variáveis explanatórias, pelo **model**.

Nesta sessão será utilizado o banco de dados originário de um ensaio clínico onde pacientes com câncer de pulmão foram alocados aleatoriamente para receber dois tipos diferentes de quimioterapia (terapia seqüencial e alternada). A variável resposta foi classificada em 4 categorias: doença progressiva, sem mudança, remissão parcial e remissão completa. Os dados foram publicados por Holtbrugge e Schumacher (1991). A análise principal será avaliar as duas terapias.

- **use c:\cursosta\tumor.dta**
- **browse**

Transformando a variável resposta em uma variável dicotômica:

- **tab resultado, nol**
- **gen resultado=resposta**
- **recode resultado 1/2=1 3/4=0**

  Portanto 1=piora e 0=melhora

Calculando os *odds* de melhora segundo terapia:

- **tabodds resultado terapia**

```
------------------------------------------------------------------------------
   terapia  |      cases     controls       odds      [95% Conf. Interval]
------------+-----------------------------------------------------------------
        seq |         89           62    1.43548        1.03798   1.98521
        alt |        104           44    2.36364        1.66150   3.36249
------------------------------------------------------------------------------
Test of homogeneity (equal odds): chi2(1)  =     4.18
                                  Pr>chi2  =   0.0409

Score test for trend of odds:     chi2(1)  =     4.18
                                  Pr>chi2  =   0.0409
```

Cuidado! O programa considera caso o valor 1 e controle o valor 0, portanto resultado=1= caso (piora) e resultado=0= controle (melhora).

Calculando o *odds ratio*:

- **mhodds resultado terapia**

```
Maximum likelihood estimate of the odds ratio
Comparing terapia==1 vs. terapia==0

    ------------------------------------------------------------------
     Odds Ratio    chi2(1)        P>chi2        [95% Conf. Interval]
    ------------------------------------------------------------------
       1.646578       4.18        0.0409         1.015699   2.669314
    ------------------------------------------------------------------
```

Lembrando: terapia 0= sequencial e terapia 1=alternada

- **cc resultado terapia**

```
                                                        Proportion
                 |   Exposed   Unexposed  |      Total     Exposed
-----------------+------------------------+------------------------
           Cases |       104          89  |        193      0.5389
        Controls |        44          62  |        106      0.4151
-----------------+------------------------+------------------------
           Total |       148         151  |        299      0.4950
                 |                        |
                 |      Point estimate    |  [95% Conf. Interval]
                 |------------------------+------------------------
      Odds ratio |         1.646578       |  .9925137    2.737841 (exact)
 Attr. frac. ex. |         .3926799       | -.0075428    .6347487 (exact)
 Attr. frac. pop |         .2115995       |
                 +-----------------------------------------------
                            chi2(1) =     4.19  Pr>chi2 = 0.0406
```

Reforçando: caso=piora, controle=melhora, exposto=alternado, não exposto= seqüencial**.**

4.4.1 - Regressão logística (logit | logistic)

- **logit resultado terapia**

```
Iteration 0:   log likelihood = -194.40888
Iteration 1:   log likelihood = -192.30753
Iteration 2:   log likelihood = -192.30471

Logit estimates                                   Number of obs   =        299
                                                  LR chi2(1)      =       4.21
                                                  Prob > chi2     =     0.0402
Log likelihood = -192.30471                       Pseudo R2       =     0.0108

------------------------------------------------------------------------------
   resultado |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     terapia |   .4986993   .2443508     2.04   0.041     .0197805     .977618
       _cons |    .361502   .1654236     2.19   0.029     .0372777    .6857263
------------------------------------------------------------------------------
```

O algoritmo precisa de 3 iterações para convergir. O coeficiente de terapia representa a diferença no *log odds* (de uma melhora) entre as terapias alternada e seqüencial. O valor negativo indica que a terapia seqüencial é superior à terapia alternada. O valor de *p* associado à estatística *z* do teste de *Wald* é 0,041. A estatística *z* é igual ao coeficiente dividido pelo erro padrão. Este valor de *p* é assintoticamente igual ao valor de *p* derivado do teste da razão de verossimilhança entre o modelo incluindo somente a constante e o modelo incluindo a variável terapia (chi2(1)=4,21). -2 vezes o logaritmo da razão de verossimilhança é igual a 4,21 com distribuição aproximada qui quadrado, com 1 grau de liberdade, com valor *p*= 0,040.

- **logistic resultado terapia**

```
Logit estimates                                   Number of obs   =        299
                                                  LR chi2(1)      =       4.21
                                                  Prob > chi2     =     0.0402
Log likelihood = -192.30471                       Pseudo R2       =     0.0108

------------------------------------------------------------------------------
   resultado | Odds Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     terapia |   1.646578   .4023427     2.04   0.041     1.019977    2.658117
------------------------------------------------------------------------------
```

- **lrtest, saving (1)**
- **logistic resultado terapia sexo**

```
Logit estimates                                   Number of obs   =        299
                                                  LR chi2(2)      =       7.55
                                                  Prob > chi2     =     0.0229
Log likelihood = -190.63171                       Pseudo R2       =     0.0194

------------------------------------------------------------------------------
   resultado | Odds Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     terapia |   1.652355   .4059667     2.04   0.041     1.020873    2.674452
        sexo |   1.923819   .7146486     1.76   0.078      .928892    3.984405
------------------------------------------------------------------------------
```

- **lrtest, saving (2)**
- **lrtest, using(2) model(1)**

```
Logistic:  likelihood-ratio test                  chi2(1)     =       3.35
                                                  Prob > chi2 =     0.0674
```

4.4.2- Regressão linear (regress)

Abrir o arquivo c:\cusrsosta\fem.dta

- **regress weight depress**

```
      Source |       SS       df       MS              Number of obs =     102
-------------+------------------------------           F(  1,   100) =    0.05
       Model |  .415589482     1  .415589482           Prob > F      =  0.8157
    Residual |  760.949608   100  7.60949608           R-squared     =  0.0005
-------------+------------------------------           Adj R-squared = -0.0094
       Total |  761.365198   101  7.53826928           Root MSE      =  2.7585

------------------------------------------------------------------------------
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     depress |    .102739   .4396238     0.23   0.816     -.769462    .9749401
       _cons |   1.295717   .8878258     1.46   0.148    -.4657039    3.057138
------------------------------------------------------------------------------
```

Ajusta um modelo de regressão linear entre weight (variável dependente) em depress (variável independente categórica).

- **tab depress, gen(depress)**

Cria variável *dummy*

- **regress weight depress2 depress3**

```
      Source |       SS       df       MS              Number of obs =     102
-------------+------------------------------           F(  2,    99) =    0.82
       Model |  12.4230796     2  6.21153982           Prob > F      =  0.4429
    Residual |  748.942118    99   7.5650719           R-squared     =  0.0163
-------------+------------------------------           Adj R-squared = -0.0036
       Total |  761.365198   101  7.53826928           Root MSE      =  2.7505

------------------------------------------------------------------------------
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    depress2 |  -.5209678   .6612327    -0.79   0.433    -1.832997    .7910612
    depress3 |     .38125   .8877096     0.43   0.669    -1.380158    2.142658
       _cons |        1.75   .5614368     3.12   0.002     .6359875    2.864012
------------------------------------------------------------------------------
```

Ajusta um modelo de regressão de weight em depress2 e depress3**,** tendo depress1 como basal (variáveis dummy; a categoria referência (depress1) não é colocada na sintaxe do comando).

Regressão linear entre 2 variáveis contínuas

- **regress weight age**

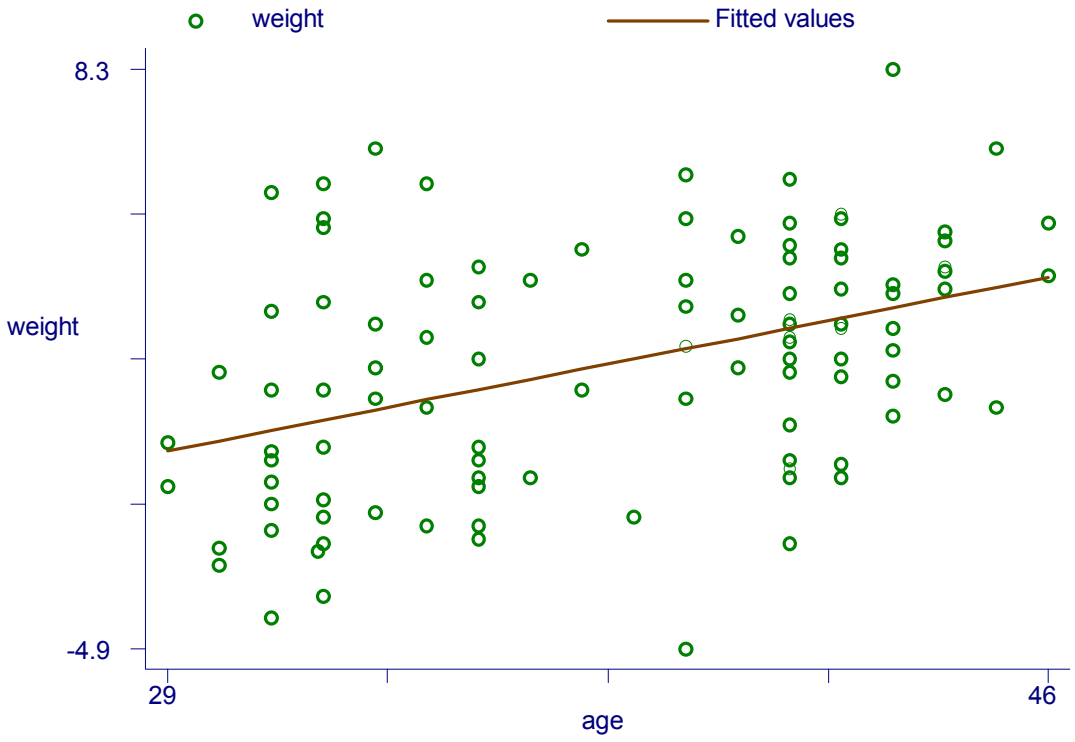
```
      Source |       SS       df       MS              Number of obs =     107
-------------+------------------------------           F(  1,   105) =   21.93
       Model |  135.142248     1  135.142248           Prob > F      =  0.0000
    Residual |   647.13383   105  6.16317933           R-squared     =  0.1728
-------------+------------------------------           Adj R-squared =  0.1649
       Total |  782.276078   106    7.379963           Root MSE      =  2.4826

------------------------------------------------------------------------------
      weight |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |    .233029   .0497642     4.68   0.000     .1343559    .3317022
       _cons |  -7.158987   1.882679    -3.80   0.000    -10.89199   -3.425981
------------------------------------------------------------------------------
```

- **predict weight2**

Cria uma nova variável que é o valor do incremento de peso ajustado pela idade.

- **graph weight weight2 age, c(.l) s(Oi) sort**

c(.l) significa "não conecte weight" e "conecte age2"; s(Oi) significa "use círculos largos para weight" e "use símbolos invisíveis para age2". O *sort* ordena os dados segundo *age*.



5- Análise de sobrevida

Pacientes com dependência a heroína, internados em uma clínica de tratamento com metadona. O evento de interesse é abandono do tratamento. Os pacientes ainda internados no término do estudo estão registrados na variável **status** (1 se o paciente abandonou o tratamento, 0 caso contrário). As variáveis explanatórias para a saída do tratamento são dose máxima de metadona, detenção prisional e clínica onde foi internado. Estes dados foram coletados e analisados por Caplehorn e Bell (1991). Variáveis estudadas:

id: identificação do paciente

clinic: clínica de internação (1, 2)

status: variável de censura (1 - abandono, 0 - em tratamento)

time: tempo de tratamento

prison: tem registro de encarceramento (1) ou não (0)

dose: dose máxima de metadona

Os dados estão disponíveis no banco c:\cursosta\heroina

5.1 - Apresentação dos dados

Declarando os dados como sendo na foram "st" (survival time)

- **stset time, failure(status)**

```
     failure event:  status ~= 0 & status ~= .
obs. time interval:  (0, time]
 exit on or before:  failure

--------------------------------------------------------------------------
      238  total obs.
        0  exclusions
--------------------------------------------------------------------------
      238  obs. remaining, representing
      150  failures in single record/single failure data
    95812  total analysis time at risk, at risk from t =         0
                             earliest observed entry t =         0
                                  last observed exit t =      1076
```

Resumindo os dados

stsum

```
failure _d:  status
   analysis time _t:  time

         |              incidence       no. of    |------ Survival time -----|
         | time at risk     rate        subjects        25%       50%       75%
---------+--------------------------------------------------------------------
   total |        95812   .0015656           238        212       504       821
```

São 238 pacientes, com tempo mediano de "sobrevida" de 504 dias. Se a taxa de incidência (hazard ratio) for constante, é estimada como 0,0016 abandonos por dia, que corresponde a 150 abandonos/95812 dias.

Pode-se realizar a análise para cada clínica:

- **strate clinic**

```
          failure _d:  status
   analysis time _t:  time

Estimated rates and lower/upper bounds of 95% confidence intervals
(238 records included in the analysis)

   clinic     _D         _Y      _Rate     _Lower     _Upper
        1    122      59558  0.0020484  0.0017154  0.0024462
        2     28      36254  0.0007723  0.0005333  0.0011186
```

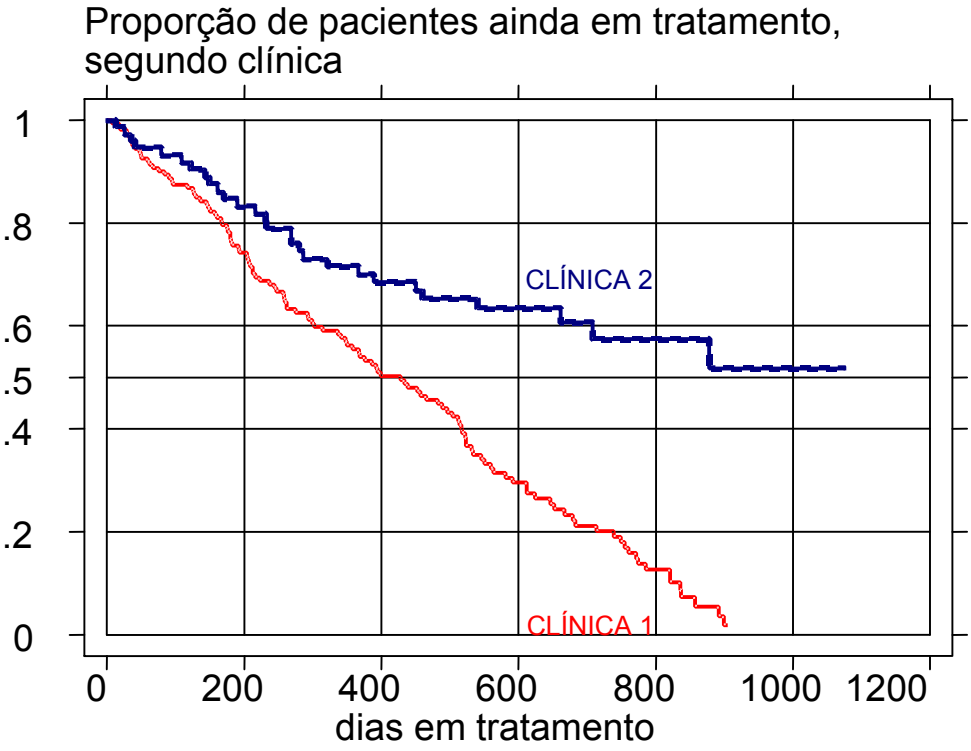
- **stsum, by(clinic)**

```
         failure _d:  status
   analysis time _t:  time

         |              incidence       no. of    |------ Survival time -----|
clinic   | time at risk     rate        subjects        25%       50%       75%
---------+--------------------------------------------------------------------
       1 |        59558   .0020484           163        192       428       652
       2 |        36254   .0007723            75        280         .         .
---------+--------------------------------------------------------------------
   total |        95812   .0015656           238        212       504       821
```

5.2- Curvas Kaplan-Meier

Construindo gráficos das curvas Kaplan-Meier

- **set textsize 150**
- **sts graph, by(clinic) xlabel(0 200 400 600 800 1000 1200) xline(0 200 400 600 800 1000 1200) ylabel(0 .2 .4 .5 .6 .8 1) yline(0 .2 .4 .5 .6 .8 1) b2(dias em tratamento)t1(Proporção de pacientes ainda em tratamento,) t2(segundo clínica)**



- **sts test clinic**

```
         failure _d:  status
   analysis time _t:  time


Log-rank test for equality of survivor functions (teste Mantel-Cox)
-----------------------------------------------

       |  Events
clinic |  observed       expected
-------+--------------------------
1      |       122          90.91
2      |        28          59.09
-------+--------------------------
Total  |       150         150.00

             chi2(1) =      27.89
             Pr>chi2 =     0.000
```

- **stcox clinic**

```
failure _d:  status
  analysis time _t:  time

Iteration 0:   log likelihood =  -705.6619
Iteration 1:   log likelihood = -690.57156
Iteration 2:   log likelihood = -690.20742
Iteration 3:   log likelihood = -690.20658
Refining estimates:
Iteration 0:   log likelihood = -690.20658

Cox regression -- Breslow method for ties

No. of subjects =        238                Number of obs   =       238
No. of failures =        150
Time at risk    =      95812
                                            LR chi2(1)      =     30.91
Log likelihood  =   -690.20658              Prob > chi2     =    0.0000

------------------------------------------------------------------------------
          _t |
          _d | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      clinic |   .3416238   .0726424    -5.05   0.000     .2251904    .5182585
------------------------------------------------------------------------------
```

5.3 - Modelo de Cox (utilizando clinicas como estrato e as outras variáveis como explanatórias)

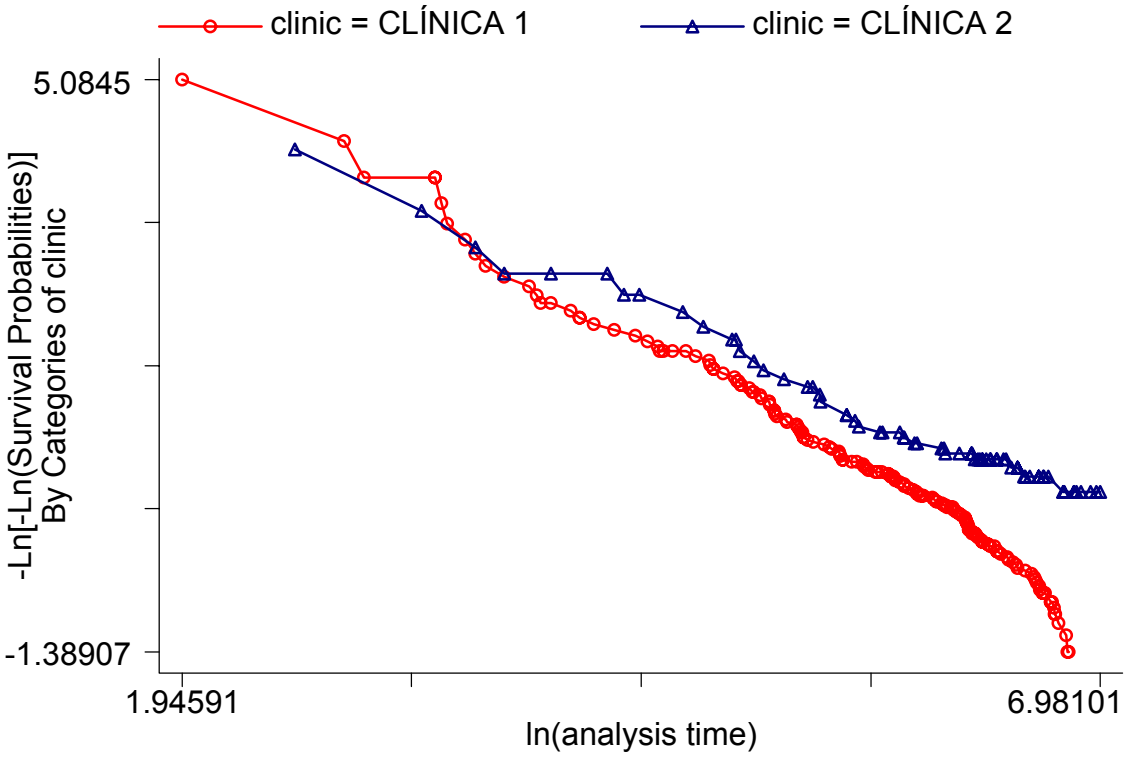
- **stcox dose prison,strata(clinic)**

```
. stcox dose prison,strata(clinic)

         failure _d:  status
   analysis time _t:  time

Iteration 0:   log likelihood = -614.68365
Iteration 1:   log likelihood = -597.73516
Iteration 2:   log likelihood =   -597.714
Refining estimates:
Iteration 0:   log likelihood =   -597.714

Stratified Cox regr. -- Breslow method for ties

No. of subjects =          238                     Number of obs   =       238
No. of failures =          150
Time at risk    =        95812
                                                   LR chi2(2)      =     33.94
Log likelihood  =     -597.714                     Prob > chi2     =    0.0000

------------------------------------------------------------------------------
      _t |
      _d | Haz. Ratio   Std. Err.       z     P>|z|       [95% Conf. Interval]
---------+--------------------------------------------------------------------
    dose |   .9654655   .0062418     -5.436   0.000        .953309     .977777
 prison  |   1.475192   .2491827      2.302   0.021       1.059418    2.054138
------------------------------------------------------------------------------
                                                        Stratified by clinic
```

Pacientes com história de prisão tendem a abandonar o tratamento mais rapidamente do que aqueles sem história de prisão. Para cada aumento de uma unidade (1 mg) na dose de metadona, o *hazard* é multiplicado por 0,965, ou seja, maior dose de metadona implica maior tempo no tratamento. Pacientes da clínica ficam mais tempo em tratamento.

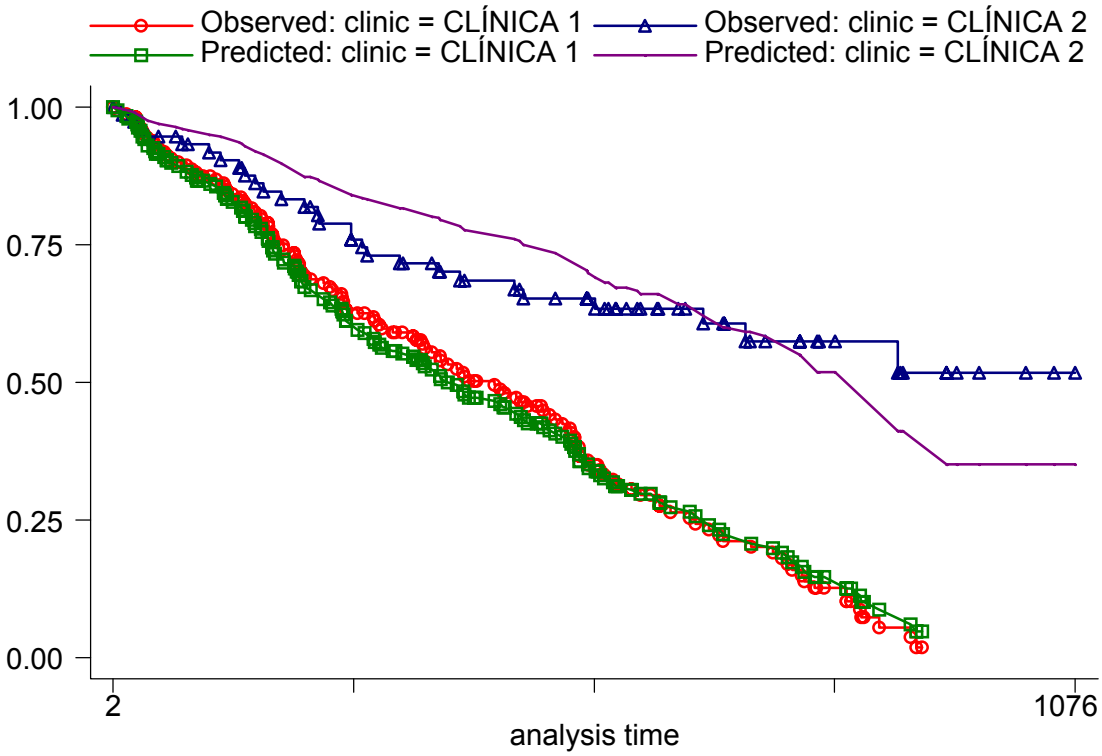
Uma questão importante é se o modelo de *hazards* proporcionais de Cox não é violado quando da comparação entre as clínicas ou da comparação entre prisioneiros e não prisioneiros. A *hazards* ratio deve ser constante no tempo.

- stphplot, by(clinic)



A análise visual indica que a proporcionalidade não se mantém no tempo.

- stcoxkm, by(clinic)



6- Comandos gerais

6.1 – Stata como calculadora

- **display <exp>**
- **display sqrt(5*((11-3)^2))**

6.2 – Breve introdução a arquivos *.do

Às vezes é necessário realizar uma análise igual para conjuntos de dados diferentes. Isto é possível, armazenando-se os comandos em um arquivo com extensão **.do**.

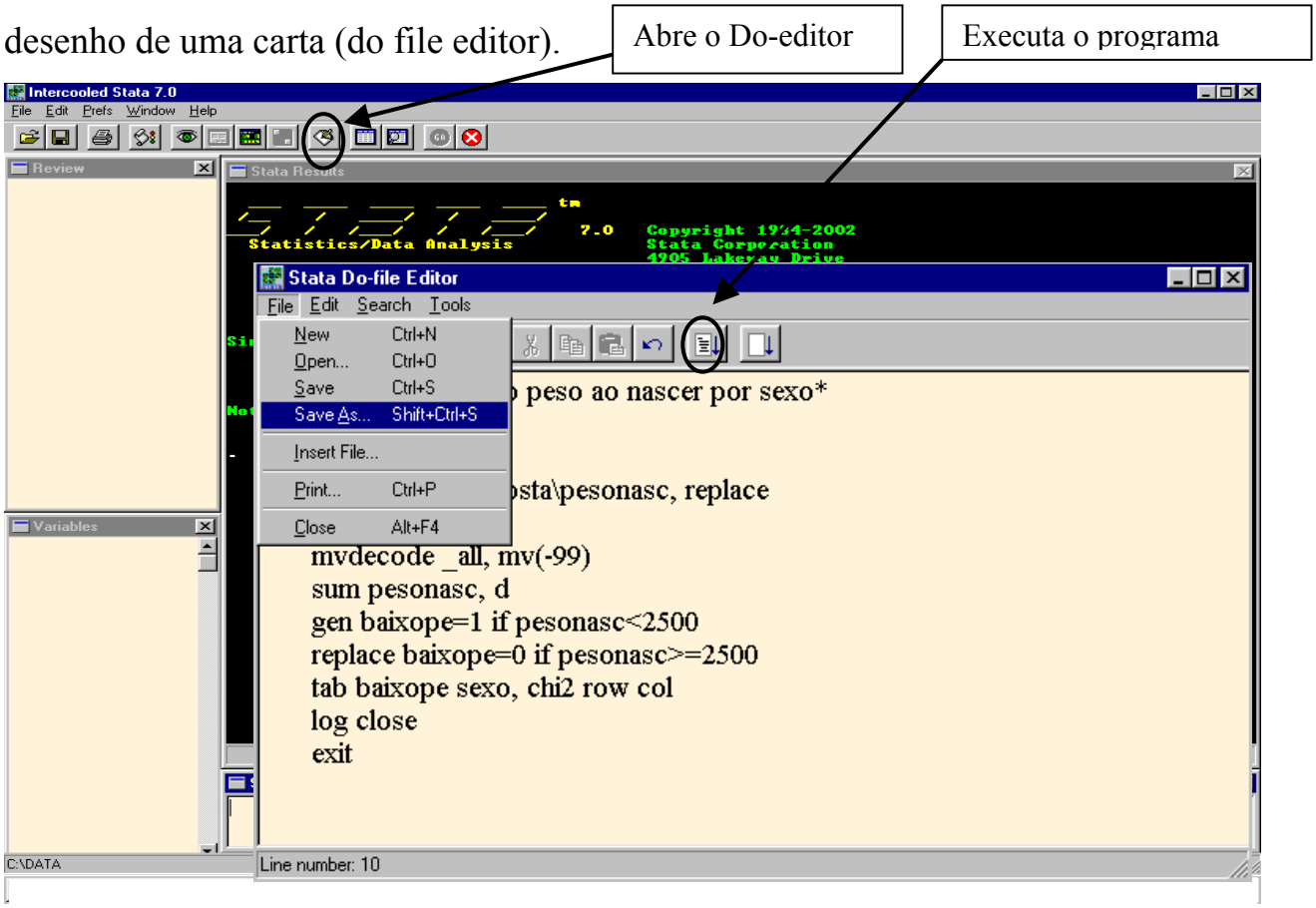
Uma forma de criar um arquivo *.do é salvando os comandos utilizados durante a sessão de trabalho. Isto pode ser feito selecionando "**save review contents**" do menu da janela "**Review**". Qualquer processador de texto pode ser utilizado para a correção dos comandos, lembrando que o arquivo *.do é texto, em ASCII. A seguir é apresentada uma estrutura básica de um arquivo *.do:

```
*comentário descrevendo o que o arquivo faz*
capture log close
log using filename, replace
set more off
command 1
command 2
.
.
log close
exit
```

Onde cada linha significa:

1. O asterisco fazem com que seja ignorado o que está entre eles; são usados para comentários.
2. O comando **capture** faz com que o Stata continue rodando mesmo que ocorra um erro na execução de um comando. O comando **capture log close** fecha o arquivo **log** em uso se for aberto outro ou envia mensagem de erro.
3. O comando **log using filename, append** abre um arquivo **log** e continua salvando as tabelas em seqüência ao já existente.
4. O comando **set more off** faz com que a saída seja apresentada na tela automaticamente, sem ter que manualmente instruir o Stata para mostrar o que está faltando.
5. Depois que a lista de comandos já estiver digitada e os resultados prontos, o arquivo **.log** é fechado com o comando **log close**.
6. A última linha do programa contendo o comando **exit** faz com que o programa pare de ser rodado.

Para abrir um arquivo .do pressionar com o *mouse* o oitavo ícone do menu, com o desenho de uma carta (do file editor).



Digitar a seqüência de comandos:

```
*Analise de baixo peso ao nascer por sexo*
capture log close
log using c:\cursosta\pesonasc, append
set more off
mvdecode _all, mv(-99)
sum pesonasc, d
gen baixope=1 if pesonasc<2500
replace baixope=0 if pesonasc>=2500
tab baixope sexo, chi2 row col
log close
exit
```

Após o término da digitação salvar com: **File, Save as...**

Para executar o programa abrir o banco de dados c:\cursosta\campinas.dta e digitar:

- **do <nome do arquivo de programação (.do)>**

Ou

Pressionar o botão *do current file* do *Do-editor.*

Esta mesma análise poderá ser feita para o banco de dados c:\cursosta\botucatu.dta.

7- Exercício 1

1- iniciar o Stata

2- abrir um arquivo **exerc1.log** no sub-diretório c:\cursosta

3- abrir banco de dados existente em **C:\cursosta\fem2.dta**

4- estudar as variáveis existentes utilizando o comando **describe**

5- alterar o banco de dados utilizando o Editor

paciente 2	age =43	anxiety =3
paciente 10	sleep=1	life= 1

quando terminar, salve as alterações (utilizando a opção **preserve**) e volte para a janela de comandos.

6- listar age

7- renomear o nome da variável *depress* para depressao

8- formatar a variável *weight* para 2 casas após a virgula

9- salvar o banco de dados como **c:\cursosta\femcorr.dta** (utilizando a opção **Save As** do menu)

10- fechar o arquivo de dados utilizando o comando **clear**

11- verificar se o arquivo **.log** continua aberto, utilizando o quarto ícone (pergaminho) e visualizando-o.

12-fechar (suspender definitivamente) o arquivo **.log**

13-abrir arquivo de dados **c:\cursosta\breast.dta**

14- abrir arquivo **exerc1.log** como continuação (append) do arquivo

15- visualizar variáveis do banco utilizando o comando **describe**

16- listar os dados utilizando o comando **list**

17- fechar o arquivo de dados utilizando o comando **clear**

18- fechar o arquivo **exerc1.log**

19- abrir arquivo c:\cursosta\rim.dta

20- abrir um arquivo **.log (rim.log)**

21- substituir os valores codificados como -99 para valores faltantes (.)

22- recodificar a variável sexo, sendo 1=0 e 0=1

23- rotular as variáveis: **id "identificacao"**; **dias "tempo ate ocorrer o obito"**; **censura "condicao do paciente no fim do estudo"**; **tratam "tratamento"**; **doador "tipo de doador"**. Verifique se os labels foram criados corretamente através do comando describe

24- definir rótulos para as categorias das variáveis

variável	codificação	
Sexo	0 – masculino	1 – feminino
Tratam	0 – sem imunossupressor	1– com imunossupressor
Doador	0– vivo	1 – cadáver

25- verificar os rótulos gerados utilizando o comando **tab** <nome da variável> (uma de cada vez)

26- pedir um resumo das variáveis utilizando o comando **summarize** ou **sum**

27- gerar uma nova variável **idade_30** centrada na média utilizando o comando **gen idade_30 = idade – 30**

28- listar as variáveis **idade** e **idade_30**; verificar se a nova variável foi criada corretamente

29- gerar uma nova variável (**catidad)** que categorize a idade em:

Faixa etária	Código
10 |-- 21	1
21 |-- 31	2
31 |-- 41	3
≥ 41	4

Cuidado: valores *missing* serão categorizados na ultima categoria se não houver uma linha de comando específica para esta situação!!!

30-definir rótulos para as categorias de **catidad**

31- tabular a variável **catidad**

32- retirar a variável **idade_30**

33- fazer o teste de associação Qui-quadrado entre as variáveis sexo e doador, com as porcentagens na linha.

34- fazer o teste de associação Exato de Fisher entre as variáveis doador e tratamento, com as porcentagens na linha e coluna.

35- fazer o teste de diferenças de duas médias ("t de Student") para idade segundo tratamento

36- salvar o banco de dados incluindo a nova variável gerada utilizando o comando **save**, **replace**

37- fechar o arquivo **rim.log** e abrir no *Word*.

Gabarito – lista de comandos

1- pelo ícone ou **Iniciar**, **Programas**, **Stata**, **Intercooled Stata**

2- clicar no quarto ícone da barra de menu, mudar diretório para **c:\cursosta**, salvar com nome **exerc1.log**, fechar janela do arquivo **.log**

3- **use c:\cursosta\fem.dta** ou pelo menu, **File, Open** e seleciona-se o arquivo **fem2.dta**, no diretório c:\cursosta

4- **describe** ou **desc**

5- utilizar o editor do Stata (10° ícone) para correção ou digitar **edit**. Após as mudanças salvar, clicando em **preserve**

6- list

7- **rename depress depressao**

8- **format weight %9.2f**

9- **File, Save As**. Salvar com o nome **femcorr.dta**

10- clear

11- clicar sobre o 4^0 ícone, escolher a 1^a. opção (**Bring log window to top**); rolar a tela do arquivo **.log**, fechar a janela do arquivo **.log**

12- clicar sobre o 4^0 ícone e selecionar a opção **Close log file**.

13- **use c:\cursosta\breast.dta** ou pelo menu, **File, Open** e seleciona-se o arquivo **breast.dta**, no diretório c:\cursosta

14- clicar no quarto ícone da barra de menu, mudar diretório para **c:\cursosta**, abrir o **exerc1.log**, fechar janela do arquivo **.log.** Escolher a opção append to existing file.

15- **describe** ou **desc**

16- **list**

17- clicar sobre o 4^0 ícone e selecionar a opção **Close log file**.

18- **clear**

19- **use c:\cursosta\rim.dta** ou pelo menu, **File, Open** e seleciona-se o arquivo **rim.dta**, no diretório c:\cursosta

20- clicar no quarto ícone da barra de menu, mudar diretório para **c:\cursosta**, salvar com nome **rim.log**, fechar janela do arquivo **.log**

21- mvdecode _ all, mv(-99)

22- recode sexo 1=0 0=1

23- **label variable id "identificacao"**

label var dias "tempo ate ocorrer o obito"

label var censura "condicao do paciente no fim do estudo"

label var tratam "tratamento"

label var doador "tipo de doador"

describe ou **desc**

24- **label define cen 0"censura" 1"falha"**

label val censura cen

label define s 0"masculino" 1"feminino"

label val sexo s

label define trat 0"sem imunossupressor" 1"com imunossupressor"

label val tratam trat

label define doa 0"vivo" 1"cadaver"

label val doador doa

25- **tab censura**

tab sexo

tab tratam

tab doador

ou **tab1 censura sexo tratam doador**

26- **sum** ou **summarize**

27- **gen idade_30=idade-30**

28- **list idade idade_30**

29- **gen catidad=1 if idade<21**

replace catidad=2 if idade>=21 & idade<31

replace catidad=3 if idade>=31 & idade<41

replace catidad=4 if idade>=41

replace catidad=. if idade==.

30- **label define catid 1 "menor que 20" 2 "20 a 30" 3 “30 a 40” 4 “maior que 40”**

label val catidad catid

31- **tab catidad**

32- drop idade_30

33- tab sexo doador, chi2 row

34- tab tratam doador, exact row col

35- ttest idade, by(tratam)

36- save, replace

37-clicar sobre o 4^0 ícone, escolher a 2^a. opção **(Close log file).**

8- Exercício 2

* Arquivo fem.dta

1. Faça o resumo da variável **weight** segundo nível de depressão (variável **depress**);

2. Faça a tabela que contém somente o peso médio e o desvio padrão da variável perda de peso (**weight**) para os níveis da variável **depress**;

3. Procure no **Help** a sintaxe do comando para realizar o *teste U de Mann-Witney*;

4. Compare as mudanças de peso segundo a variável **depress**, utilizando o *teste U de Mann-Witney*;

5. Faça um histograma da variável **age** e salve-o em um arquivo **doc**.

6. Faça um **boxplot** da variável **weight** segundo níveis da variável depress.

7. Transporte este gráfico para o *Word.*

Gabarito - exercício 2

1- use "C:\cursosta\fem.dta", clear

sort depress

by depress: sum weight

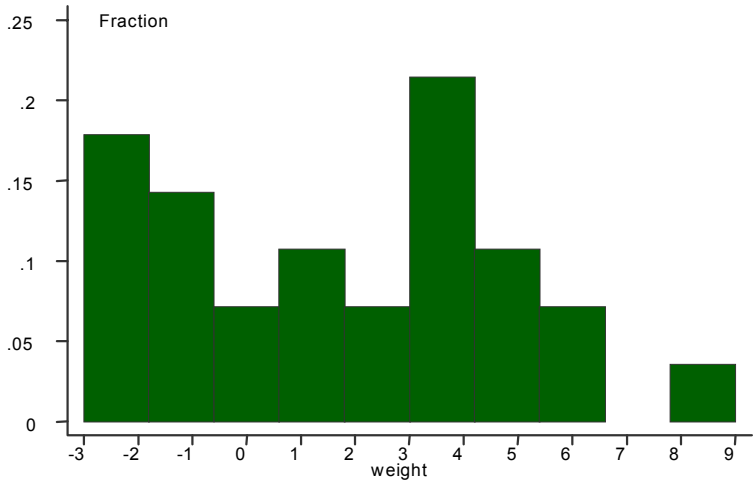
2- table depress, contents(mean weight sd weight)

3- Help, Contents. Em "Command:", digitar Mann-Whitney. Clicar na opção sign-rank (o teste de Mann-Whitney é feito pelo comando ranksum).

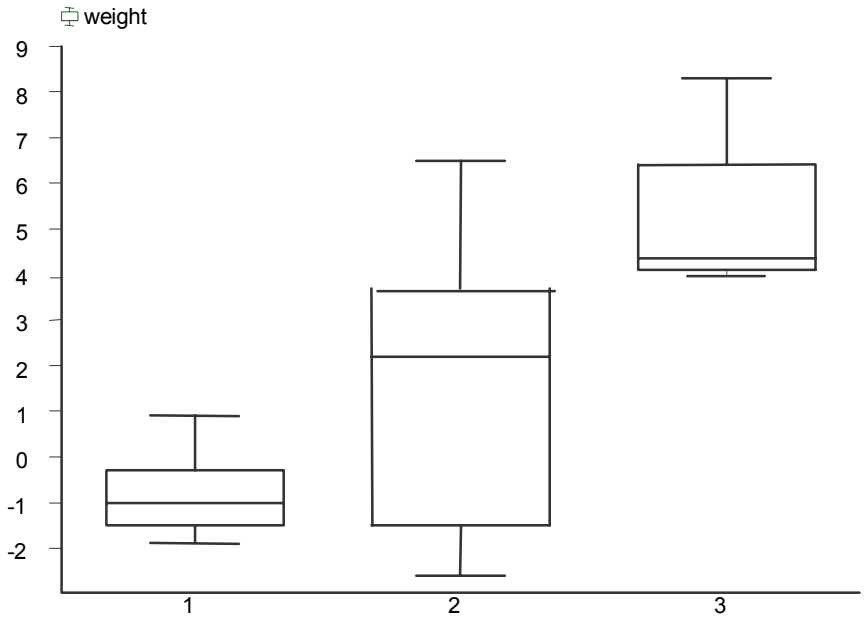
4- ranksum weight, by(life)

5- graph weight, bin(10) xlab(-3,-2,-1,0,1,2,3,4,5,6,7,8,9) ylab (0, 0.05, 0.10, 0.15, 0.20, 0.25)

Edit, Copy Graph. Abrir o Word, colar no documento e salvá-lo em um arquivo do Word.



6- graph weight, by(depressi) box ylab(-2,-1,0,1,2,3,4,5,6,7,8,9)



Edit, Copy Graph. Abrir o Word, colar no documento e salvá-lo em um arquivo do Word.

9- Bibliografia

Caplehorn J e Bell J. Methadone dosage and the retention of patients in maintenance treatment. *The medical Journal of Australia*, 154:195-9, 1991.

Conrad S. *Assignments in Applied Statistics*. Wiley, Chichester, 1989 (p.126).

Hamilton LC. *Statistics with Stata 5*. Duxbury Press, Belmont, CA, 1998.

Hand DJ et al. *A Handbook of Samall Data Sets*. Chapmen e Hall, London, 1994.

Holtbrugge W e Schumacher M. A comparison of regression models for the analysis of ordered categorical data. *Applied Statistics*, 40:249-59, 1991.

Lea AJ New observations on distribution of neoplasms of female breast in certain European countries. *British Medical Journal*, 1, 488-490, 1965.

Mazess RB; Peppler WW & Gibbons M Total body composition by dual-photon (^{153}Gd) absorptiometry. *American Journal of Clinical Nutrition*, 40, 834-839, 1984.

StataCorp Stata Statistical Software: release 7.0. Stata Corporation, 2001 .