【实训内容】

（1）启动虚拟机（在虚拟机上将安装镜像文件加载到虚拟光驱），运行安装向导，选择

“中文（简体）”。

（2）单击“安装 Ubuntu”按钮，选择键盘布局，可选择“英语（美国）”。

（3）单击“继续”按钮，选择“正常安装”和“安装 Ubuntu 时下载更新”。

（4）单击“继续”按钮，选择第 1 种类型“清除整个磁盘并安装 Ubuntu”。

（5）单击“现在安装”按钮，确认将改动写入磁盘。

（6）单击“继续”按钮，选择所在时区，默认值为“Shanghai”。

（7）单击“继续”按钮，输入个人姓名和计算机名。

（8）单击“继续”按钮，进入正式的安装界面，安装过程中需要在线下载软件包。

（9）安装完成后，根据提示移除安装介质并重启计算机。

（10）重启后即可登录 Ubuntu 系统。

实训目的： 掌握 Ubuntu Linux 操作系统的安装。

实训准备：

ISO 镜像文件： 建议下载最新版本的 Ubuntu 桌面版 ISO 镜像文件。也可使用教材配套版本 Ubuntu21.04-desktop (下载地址：http://releases.ubuntu.com/21.04/)。

实验用计算机： 可使用 VMware Workstation 或 Virtualbox 虚拟机软件创建 Ubuntu Linux 虚拟机。内存建议 4GB 以上，硬盘和网络模式等其他选项可选择默认值。

实训内容：

启动虚拟机： 在虚拟机中将安装镜像文件加载到虚拟光驱，运行安装向导。

选择语言： 选择“中文（简体）”。

安装 Ubuntu： 单击“安装 Ubuntu”按钮。

键盘布局： 选择键盘布局，例如“英语（美国）”。

安装类型： 选择“正常安装”和“安装 Ubuntu 时下载更新”。

分区方案： 选择“清除整个磁盘并安装 Ubuntu”。

写入磁盘： 单击“现在安装”按钮，确认将改动写入磁盘。

时区选择： 选择所在时区，默认值为“Shanghai”。

用户信息： 输入个人姓名和计算机名。

开始安装： 单击“继续”按钮，进入正式安装界面。

移除安装介质： 安装完成后，根据提示移除安装介质并重启计算机。

登录系统： 重启后即可登录 Ubuntu 系统。

【实训内容】

（1）使用活动概览视图。屏幕左上角“活动”（activities）按钮，或者键盘上的 windows

2

键。

（2）启动应用程序、熟悉窗口操作。

（3）使用文件管理器。

（4）使用 gedit 编辑器。

（5）使用 office 系列应用程序

（6）使用软件中心安装和卸载软件。

（7）进行桌面个性化设置，如：调整背景图片，分辨率等。

（8）使用 ctrl+alt+t 打开终端窗口，在界面中分别输入 ls、pwd、exit 命令并回车执行。

实训目的：

熟悉 Ubuntu 桌面环境和基本操作。

熟悉常用软件的使用。

实训准备： 已安装好 Ubuntu 桌面操作系统。

实训内容：

活动概览视图：

点击屏幕左上角的“活动”（activities）按钮，或按下键盘上的 windows 键，进入活动概览视图。

熟悉界面布局，包括搜索框、应用启动器、工作区切换按钮等。

启动应用程序：

使用搜索框搜索并启动应用程序，例如火狐浏览器、文本编辑器等。

熟悉窗口操作，包括最大化、最小化、关闭、移动窗口等。

文件管理器：

打开文件管理器，查看文件和目录结构。

创建、删除、重命名文件和目录。

复制、移动文件和目录。

设置文件和目录的权限。

gedit 编辑器：

使用 gedit 编辑器打开和编辑文本文件。

熟悉 gedit 的基本操作，例如插入文本、删除文本、查找和替换等。

Office 系列应用程序：

使用 LibreOffice 套件中的文字处理、表格处理、演示文稿等应用程序。

创建、编辑和保存文档、表格、演示文稿等。

软件中心：

打开软件中心，浏览和安装各种应用程序。

使用软件中心搜索和安装软件。

使用软件中心卸载软件。

桌面个性化设置：

调整背景图片、窗口主题、字体等。

调整分辨率、屏幕亮度等。

终端窗口：

使用 ctrl+alt+t 快捷键打开终端窗口。

在终端中输入 ls 命令，列出当前目录下的文件和目录。

输入 pwd 命令，查看当前工作目录的路径。

输入 exit 命令，退出终端窗口。

【实训内容】

（1）打开终端窗口。

（2）运行命令 sudo apt intstall open-vm-tools 安装基础包(需要输入密码)。比较安装前

后的界面效果的变化。

（3）运行命令 sudo apt intstall open-vm-tools-desktop。可支持桌面环境文件拖放。

实训目的： 安装 VM-tools 工具，提高虚拟机性能和易用性。

实训准备： 已安装好 Ubuntu 桌面操作系统。

实训内容：

打开终端窗口： 使用 ctrl+alt+t 快捷键打开终端窗口。

安装基础包：

输入命令 sudo apt install open-vm-tools，安装 VM-tools 基础包。

根据提示输入密码，并确认安装。

安装完成后，可以观察到界面效果的变化，例如分辨率调整、鼠标操作更流畅等。

安装桌面工具：

输入命令 sudo apt install open-vm-tools-desktop，安装 VM-tools 桌面工具。

该工具支持桌面环境文件拖放等功能，进一步提升易用性。

注意事项：

安装 VM-tools 前需要确保虚拟机操作系统版本与 VM-tools 版本兼容。

安装过程中需要输入管理员密码，确认操作。

安装完成后，部分功能可能需要重启虚拟机才能生效。

【实训内容】

（1）执行 ls 命令；

（2）执行带-l 选项的 ls 命令；执行带参数的 ls 命令。

（3）调用历史命令。完成教材 3.4.4 节实例

（4）命令补齐功能。

用 date 命令查看系统当前时间，在输入 da 后，按 Tab 键，让 Shell 自动补齐命令的后

半部分。

用 mkdir 命令创建新的目录。首先输入第一个字母 m，然后按 Tab 键，由于以 m 开头

的命令太多，Shell 会提示是否显示全部的可能命令，输入 n。

3

再多输入一个字母 k，按 Tab 键，让 Shell 列出以 mk 开头的所有命令的列表。

在列表中查找 mkdir 命令，看看还需要多输入几个字母才能确定 mkdir 这个命令，然

后输入需要的字母，再按 Tab 键，让 Shell 补齐剩下的命令。

最后输入要创建的目录名，按 Enter 键执行命令。

多试几个命令利用 Tab 键补齐。

（5）命令续行。完成实例 3-22。

（6）强制中断命令运行。运行 ping www.baidu.com ，强行中断之。

（7）获得联机帮助。使用帮助系统，查找 ls 的帮助信息。

（8）输入重定向。完成教材实例 3-19 和 3-20。

（9）输出重定向。

用 ls 命令显示当前目录中的文件列表。

使用输出重定向，把 ls 命令在终端上显示的当前目录中的文件列表重定向到文件 zpfile

中。

查看文件 zpfile 中的内容，注意在列表中会多出一个文件 zpfile。

然后再运行 ls 命令，再次使用输出重定向，把 ls 命令在终端上显示的当前目录中的文

件列表重定向到文件 zpfile 中。这次使用管道符号>>进行重定向。

查看文件 zpfile 的内容，可以看到用>>进行重定向是把新的输出内容附加在文件的末尾，

注意其中两行 zpfile 文件的信息中文件大小的区别。

（10）管道操作。

利用管道和 grep 命令，在上面建立的文件 zpfile 中查找字符串 zpfile。

利用管道和 wc 命令，计算文件 zpfile 中的行数、单词数和字符数。

（11）命令替换。

完成教材实例 3-23

（12）命令别名。

输入 alias 命令，显示目前已经设置好的命令的别名。

 设置别名 ls 为 ls –l，以长格式显示文件列表。

 显示别名 ls 代表的命令，确认设置生效。

 使用别名 ls 显示当前目录中的文件列表。

 在使定义的别名不失效的情况下，使用系统的 ls 命令显示当前目录中的命令列表。

 删除别名。

 显示别名 ls，确认删除别名已经生效。

 用命令 ls 显示当前目录中的文件列表。

实训目的：

熟悉命令语法格式。

熟悉命令行基本用法。

实训准备： 已安装好 Ubuntu 桌面操作系统。

实训内容：

执行 ls 命令：

在终端中输入 ls 命令，列出当前目录下的文件和目录。

执行带参数的 ls 命令：

输入 ls -l 命令，以长格式显示文件和目录信息，包括权限、大小、创建时间等。

尝试使用其他参数，例如 -a (显示隐藏文件)、-h (以可读格式显示文件大小) 等。

调用历史命令：

使用 history 命令查看历史命令列表。

使用 ! 符号加命令编号，重新执行历史命令。

例如，输入 !10 将重新执行编号为 10 的历史命令。

命令补齐功能：

输入命令的前几个字母，然后按 Tab 键，Shell 会自动补齐剩余部分。

如果存在多个匹配项，可以多次按 Tab 键查看所有选项。

命令续行：

使用反斜杠 \ 将命令续行到下一行。

例如，输入 echo Hello \ World 将输出 Hello World。

强制中断命令运行：

使用 Ctrl + C 组合键强制中断正在运行的命令。

例如，运行 ping www.baidu.com 后，按下 Ctrl + C 将停止 ping 操作。

获得联机帮助：

使用 man 命令查看命令的详细说明。

例如，输入 man ls 将显示 ls 命令的说明文档。

输入重定向：

使用 < 符号将文件内容作为命令的输入。

例如，输入 wc < file.txt 将统计 file.txt 文件中的行数、单词数和字符数。

输出重定向：

使用 > 符号将命令的输出保存到文件中。

例如，输入 ls > file.txt 将当前目录下的文件列表保存到 file.txt 文件中。

使用 >> 符号将命令的输出追加到文件末尾。

管道操作：

使用 | 符号将一个命令的输出作为另一个命令的输入。

例如，输入 ls | grep "file" 将列出包含 “file” 字符串的文件名。

命令替换：

使用 $(command) 将命令的输出替换到当前位置。

例如，输入 echo $(date) 将显示当前日期和时间。

命令别名：

使用 alias 命令创建命令别名。

例如，输入 alias ll='ls -l' 将创建 ll 别名，用于显示长格式文件列表。

使用 unalias 命令删除别名。

注意事项：

命令行操作时，注意区分大小写。

使用 Tab 键补齐命令时，需要根据实际情况调整输入的字母数量。

使用 Ctrl + C 强制中断命令时，请确保不会影响系统正常运行。

【实训内容】

 打开终端窗口。

 在当前目录下，创建测试目录 test。

#mkdir test

 利用 ls 命令列出文件和目录，确认 test 目录创建成功。

#ls

 进入 test 目录，利用 pwd 命令查看当前工作目录。

#cd test

#pwd

 利用 touch 命令，在当前目录创建一个新的空文件 newfile。

#touch newfile

 利用 cp 命令复制系统文件/etc/profile 到当前目录下。

#cp /etc/profile

 复制文件 profile 到一个新文件 profile.bak，作为备份。

#cp profile profile.bak

 用 ll 命令以长格式列出当前目录下的所有文件，注意比较每个文件的长度和创建时

间的不同。

#ll

 用 less 命令分屏查看文件 profile 的内容，注意练习 less 命令的各个子命令，例如

b、p、q 等并对 then 关键字查找。

#less profile

 用 grep 命令在 profile 文件中对关键字 then 进行查询，并与上面的结果比较。

#grep then profile

 给文件 profile 创建一个符号链接 lnsprofile 和一个硬链接 lnhprofile。

# ln -s profile lnsprofile

5

# ln profile lnhprofile

 长格式显示文件 profile、lnsprofile 和 lnhprofile 的详细信息。注意比较 3 个文件链

接数的不同。

#ll

 删除文件 profile，用长格式显示文件 lnsprofile 和 lnhprofile 的详细信息，比较文件

lnhprofile 的链接数的变化。

 用 less 命令查看文件 lnsprofile 的内容，看看有什么结果。

 用 less 命令查看文件 lnhprofile 的内容，看看有什么结果。

 删除文件 lnsprofile，显示当前目录下的文件列表，回到上层目录。

 用 tar 命令把目录 test 打包。

 用 gzip 命令把打好的包进行压缩。

 把文件 test.tar.gz 改名为 backup.tar.gz。

 显示当前目录下的文件和目录列表，确认重命名成功。

 把文件 backup.tar.gz 移动到 test 目录下。

 显示当前目录下的文件和目录列表，确认移动成功。

 进入 test 目录，显示目录中的文件列表。

 把文件 test.tar.gz 解包。

 显示当前目录下的文件和目录列表，复制 test 目录为 testbak 目录作为备份。

 查找 root 用户自己的主目录下的所有名为 newfile 的文件。

 删除 test 子目录下的所有文件。

 利用 rmdir 命令删除空子目录 test。

 回到上层目录，利用 rm 命令删除目录 test 和其下所有文件。

实训目的：

熟悉各类文件操作命令。

掌握基于命令行的文件操作。

熟悉目录操作命令。

掌握基于命令行的目录操作。

实训准备： 已安装好 Ubuntu 桌面操作系统。

实训内容：

打开终端窗口： 使用 ctrl+alt+t 快捷键打开终端窗口。

创建测试目录：

输入 mkdir test 命令，创建名为 test 的目录。

使用 ls 命令列出当前目录下的文件和目录，确认 test 目录创建成功。

进入 test 目录：

输入 cd test 命令，进入 test 目录。

使用 pwd 命令查看当前工作目录的路径。

创建空文件：

输入 touch newfile 命令，创建名为 newfile 的空文件。

复制文件：

输入 cp /etc/profile 命令，复制 /etc/profile 文件到当前目录下。

输入 cp profile profile.bak 命令，将 profile 文件复制为 profile.bak 文件，作为备份。

查看文件信息：

输入 ll 命令，以长格式列出当前目录下的所有文件，比较文件大小和创建时间。

查看文件内容：

输入 less profile 命令，分屏查看 profile 文件的内容。

使用 b (回滚一页)、p (前进一页)、q (退出) 等子命令进行操作。

使用 /then 查找包含 “then” 关键字的行。

搜索文件内容：

输入 grep "then" profile 命令，在 profile 文件中搜索包含 “then” 关键字的行。

创建链接：

输入 ln -s profile lnsprofile 命令，创建 profile 文件的符号链接 lnsprofile。

输入 ln profile lnhprofile 命令，创建 profile 文件的硬链接 lnhprofile。

查看链接信息：

输入 ll 命令，查看 profile、lnsprofile 和 lnhprofile 文件的详细信息，比较链接数。

删除文件：

输入 rm profile 命令，删除 profile 文件。

输入 ll 命令，查看 lnsprofile 和 lnhprofile 文件的链接数变化。

查看链接内容：

输入 less lnsprofile 命令，查看 lnsprofile 的内容，由于符号链接指向的文件已删除，将无法查看内容。

输入 less lnhprofile 命令，查看 lnhprofile 的内容，由于硬链接指向的文件实际存在，可以正常查看内容。

删除符号链接：

输入 rm lnsprofile 命令，删除 lnsprofile 符号链接。

打包目录：

输入 tar -cvf test.tar test 命令，将 test 目录打包为 test.tar 文件。

压缩文件：

输入 gzip test.tar 命令，将 test.tar 文件压缩为 test.tar.gz 文件。

重命名文件：

输入 mv test.tar.gz backup.tar.gz 命令，将 test.tar.gz 文件重命名为 backup.tar.gz。

移动文件：

输入 mv backup.tar.gz test 命令，将 backup.tar.gz 文件移动到 test 目录下。

解包文件：

进入 test 目录，输入 tar -xvf backup.tar.gz 命令，将 backup.tar.gz 文件解包。

复制目录：

输入 cp -r test testbak 命令，将 test 目录复制为 testbak 目录。

查找文件：

输入 find /root -name "newfile" 命令，查找 /root 目录下所有名为 newfile 的文件。

删除文件：

输入 rm -rf test/\* 命令，删除 test 子目录下的所有文件。

删除空目录：

输入 rmdir test 命令，删除空目录 test。

删除非空目录：

输入 rm -rf testbak 命令，删除非空目录 testbak 及其下所有文件。

注意事项：

操作文件和目录时，注意文件路径和权限。

使用 rm 命令删除文件时，请谨慎操作，避免误删重要文件。

使用 find 命令查找文件时，可以根据需要调整搜索范围和条件。

【实训内容】

（1）创建一个普通用户。

6

（2）修改用户的 UID。

（3）修改用户的密码和有效期等信息。

（4）创建系统用户。

（5）查看用户配置文件/etc/passwd 和/etc/shadow，观察内容变化情况。可以在命令行

中执行文件显示命令，也可以使用 grep 命令来查找。

实训目的： 掌握用户账户的命令行操作。

实训内容：

创建普通用户：

输入 sudo adduser username 命令，创建名为 username 的普通用户，并根据提示设置用户密码和其他信息。

修改用户的 UID：

输入 sudo usermod -u new\_uid username 命令，将 username 用户的 UID 修改为 new\_uid。

修改用户的密码和有效期等信息：

输入 sudo passwd username 命令，根据提示设置 username 用户的密码。

输入 sudo chage -M maxdays username 命令，设置 username 用户的密码有效期，其中 maxdays 表示密码有效期天数。

输入 sudo chage -I inactivedays username 命令，设置 username 用户的账户失效天数，即密码过期后多少天账户被禁用。

创建系统用户：

输入 sudo useradd -r system\_username 命令，创建名为 system\_username 的系统用户，其中 -r 选项表示创建系统用户。

查看用户配置文件：

输入 cat /etc/passwd 命令，查看 /etc/passwd 文件，该文件记录了系统中所有用户的信息，包括用户名、UID、GID、家目录、登录shell等。

输入 cat /etc/shadow 命令，查看 /etc/shadow 文件，该文件记录了系统中所有用户的密码信息，由于安全性原因，密码信息经过加密处理。

注意事项：

操作用户账户时，需要使用 sudo 命令获取管理员权限。

修改用户 UID 和密码时，请谨慎操作，避免影响系统正常运行。

/etc/passwd 和 /etc/shadow 文件是系统文件，请勿随意修改。

【实训步骤】

（1）创建一个新的组。

（2）修改组账户名称和 GID。

（2）查看用户所属组。

（3）将用户添加到新建组中。

（4）将用户从该新建组中删除。

（5）查看组配置文件/etc/group 和/etc/gshadow 获取组账户信息列表，观察变化情况。

实训目的： 掌握组账户的命令行操作。

实训步骤：

创建新组：

输入 sudo groupadd newgroup 命令，创建名为 newgroup 的新组。

修改组账户名称和 GID：

输入 sudo groupmod -n newgroupname newgroup 命令，将 newgroup 组的名称修改为 newgroupname。

输入 sudo groupmod -g new\_gid newgroupname 命令，将 newgroupname 组的 GID 修改为 new\_gid。

查看用户所属组：

输入 groups username 命令，查看 username 用户所属的组列表。

输入 id -Gn username 命令，查看 username 用户所属的组名称列表。

将用户添加到新建组中：

输入 sudo usermod -aG newgroupname username 命令，将 username 用户添加到 newgroupname 组中。

将用户从组中删除：

输入 sudo gpasswd -d username newgroupname 命令，将 username 用户从 newgroupname 组中删除。

查看组配置文件：

输入 cat /etc/group 命令，查看 /etc/group 文件，该文件记录了系统中所有组的信息，包括组名、GID、组成员等。

输入 cat /etc/gshadow 命令，查看 /etc/gshadow 文件，该文件记录了系统中所有组的密码信息，由于安全性原因，密码信息经过加密处理。

注意事项：

操作组账户时，需要使用 sudo 命令获取管理员权限。

修改组名称和 GID 时，请谨慎操作，避免影响系统正常运行。

/etc/group 和 /etc/gshadow 文件是系统文件，请勿随意修改。

【实训内容】

某公司有 6 个部分，每个部分都有一些员工，每个人工作内容不同。需要在服务器上为

每个人创建不同的账号，把相同部门的用户放在一个组中，每个用户都有自己的工作目录。

实训目的： 掌握在 Linux 系统中增加、修改、删除用户或用户组的方法，以及用户账户管理方法。

实训内容：

实训内容：

1. 创建用户组

使用 groupadd 命令创建 6 个用户组，例如：

复制

groupadd dept1

groupadd dept2

groupadd dept3

groupadd dept4

groupadd dept5

groupadd dept6

2. 创建用户

使用 useradd 命令创建用户，并为每个用户指定所属的用户组，例如：

复制

useradd -m -g dept1 user1

useradd -m -g dept2 user2

useradd -m -g dept3 user3

useradd -m -g dept4 user4

useradd -m -g dept5 user5

useradd -m -g dept6 user6

-m 参数表示创建用户的同时创建用户的家目录。

-g 参数表示指定用户所属的用户组。

3. 设置用户密码

使用 passwd 命令设置用户的密码，例如：

复制

passwd user1

passwd user2

passwd user3

passwd user4

passwd user5

passwd user6

4. 修改用户信息

使用 usermod 命令修改用户信息，例如：

复制

usermod -c "Department 1" user1

usermod -c "Department 2" user2

usermod -c "Department 3" user3

usermod -c "Department 4" user4

usermod -c "Department 5" user5

usermod -c "Department 6" user6

-c 参数表示修改用户的注释信息，例如部门名称。

5. 删除用户

使用 userdel 命令删除用户，例如：

复制

userdel user1

userdel user2

6. 删除用户组

使用 groupdel 命令删除用户组，例如：

复制

groupdel dept1

groupdel dept2

注意事项：

可以使用 id 命令查看用户或用户组的 ID。

可以使用 chown 命令修改文件或目录的所属用户和用户组。

可以使用 chmod 命令修改文件或目录的权限。

【实训内容】

通过 fdisk 的交互模式创建分区。分区方案如下图。其中 sdb1 和 sdb2 为主分区，sdb5

和 sdb6 为逻辑分区，各自尺寸自拟。

由于您提供的图片无法显示，我无法直接给出分区方案的具体尺寸。但是，我可以提供使用 fdisk 命令创建分区的步骤，您可以参考这些步骤并根据您的需求调整分区尺寸。

实训目的： 熟悉磁盘分区操作

实训准备： 添加一个空白硬盘（在虚拟机上添加虚拟硬盘）。

实训内容：

打开终端窗口。

输入 sudo fdisk /dev/sdb 命令，启动 fdisk 程序，其中 /dev/sdb 是您要分区的硬盘设备名。

在 fdisk 程序中，使用以下命令创建分区：

n：创建新分区

p：创建主分区 (Primary Partition)

1：指定主分区编号为 1

指定分区起始扇区 (默认为 2048)

指定分区结束扇区 (例如：+10G 表示创建 10GB 大小的分区)

重复以上步骤，创建第二个主分区 sdb2

n：创建新分区

e：创建扩展分区 (Extended Partition)

3：指定扩展分区编号为 3

指定扩展分区起始扇区 (默认为上一个分区结束扇区)

指定扩展分区结束扇区 (例如：+20G 表示创建 20GB 大小的扩展分区)

n：创建新分区

l：创建逻辑分区 (Logical Partition)

5：指定逻辑分区编号为 5

指定逻辑分区起始扇区 (默认为 2048)

指定逻辑分区结束扇区 (例如：+5G 表示创建 5GB 大小的逻辑分区)

重复以上步骤，创建第二个逻辑分区 sdb6

w：保存分区表并退出 fdisk 程序

注意事项：

在创建分区时，请根据您的需求调整分区类型和尺寸。

使用 fdisk 命令进行分区操作时，请谨慎操作，避免数据丢失。

创建完分区后，可以使用 fdisk -l /dev/sdb 命令查看分区表，确认分区创建成功。

【实训内容】

（1）使用 mkfs 命令在上述某个分区上建立 ext4 文件系统。

（2）用 fsck 检查文件系统

（3）创建一个挂载点目录。

（4）使用 mount 命令将该分区挂载到此目录。

（5）通过挂载点进入分区并新建文件和文件夹。

（6）卸载该分区，并重新挂载到一个新的位置。

（7）观察之前创建的文件和文件夹是否存在。

实训目的： 熟悉建立和使用文件系统的步骤，掌握基于命令行的文件系统操作。

实训准备： 完成上一个实训任务“磁盘分区管理”。

实训内容：

使用 mkfs 命令创建 ext4 文件系统：

输入 sudo mkfs.ext4 /dev/sdb1 命令，在 /dev/sdb1 分区上创建 ext4 文件系统。

使用 fsck 命令检查文件系统：

输入 sudo fsck.ext4 -n /dev/sdb1 命令，检查 /dev/sdb1 分区的文件系统，-n 选项表示仅进行检测，不修复问题。

创建挂载点目录：

输入 sudo mkdir /mnt/sdb1 命令，创建名为 /mnt/sdb1 的挂载点目录。

使用 mount 命令挂载分区：

输入 sudo mount /dev/sdb1 /mnt/sdb1 命令，将 /dev/sdb1 分区挂载到 /mnt/sdb1 目录。

通过挂载点进入分区并新建文件和文件夹：

输入 cd /mnt/sdb1 命令，进入挂载点目录。

输入 touch newfile 命令，创建名为 newfile 的文件。

输入 mkdir newdir 命令，创建名为 newdir 的文件夹。

卸载分区：

输入 sudo umount /mnt/sdb1 命令，卸载 /dev/sdb1 分区。

重新挂载分区到新位置：

输入 sudo mkdir /mnt/sdb1\_backup 命令，创建新的挂载点目录。

输入 sudo mount /dev/sdb1 /mnt/sdb1\_backup 命令，将 /dev/sdb1 分区挂载到 /mnt/sdb1\_backup 目录。

观察之前创建的文件和文件夹是否存在：

输入 ls /mnt/sdb1\_backup 命令，查看 /mnt/sdb1\_backup 目录下的文件和文件夹，确认 newfile 和 newdir 是否存在。

注意事项：

在执行 fsck 命令时，请根据实际情况选择是否进行修复操作。

挂载点目录必须存在才能进行挂载操作。

卸载分区前，请确保没有程序正在使用该分区。

【实训内容】

（1）使用 ps 命令监控后台进程的工作情况，尝试 aux 选项组合的使用。

（2）ps 命令结合管道操作符和 less（more）命令查看进程。

（3）ps 命令结合管道操作符和 grep 命令查看指定进程，如 python 或者 vmtool。

（4）使用 top 命令动态显示系统进程信息。

（5）使用 top 命令检测指定进程状态信息，如 python 或者 vmtool。

实训目的： 熟悉 Linux 进程参数，掌握 ps 命令和 top 命令的使用。

实训内容：

使用 ps 命令监控后台进程：

输入 ps aux 命令，查看系统中所有进程的详细信息，包括进程 ID、用户、CPU 使用率、内存使用率等。

可以使用 ps aux | grep python 命令，查看所有 python 进程的详细信息。

可以使用 ps -ef 命令，查看进程的详细信息，包括进程 ID、父进程 ID、启动时间等。

ps 命令结合管道操作符和 less (或 more) 命令查看进程：

输入 ps aux | less 命令，使用 less 命令分页查看进程信息。

可以使用 ps aux | more 命令，使用 more 命令分页查看进程信息。

ps 命令结合管道操作符和 grep 命令查看指定进程：

输入 ps aux | grep python 命令，查看所有 python 进程的详细信息。

输入 ps aux | grep vmtool 命令，查看所有 vmtool 进程的详细信息。

使用 top 命令动态显示系统进程信息：

输入 top 命令，进入 top 程序界面，动态显示系统中进程的实时信息，包括进程 ID、用户、CPU 使用率、内存使用率等。

可以使用 top -u username 命令，查看指定用户的进程信息。

使用 top 命令检测指定进程状态信息：

在 top 程序界面中，输入 p 键，然后输入进程 ID，可以查看指定进程的详细信息，包括进程状态、CPU 使用率、内存使用率等。

可以使用 top -H -p pid 命令，查看指定进程 ID 的进程的详细信息。

注意事项：

ps 命令和 top 命令可以提供丰富的进程信息，方便您监控系统进程状态。

可以使用 man ps 和 man top 命令查看 ps 命令和 top 命令的详细使用说明。

【实训内容】

（1）执行 vi &。

（2）ps 命令查看进程 vi 的 pid。

（3）使用 kill 结束 vi 进程。

（4）使用 nice vi &启动进程，观察其 niceness 值。

（5）使用 nice vi &并添加合适的参数，使得该进程 niceness 值为负数

（5）使用 ps 观察前述 vi 进程的 pid 和 niceness。

（6）使用 renice 分别调大和调小现有的 vi 进程的 niceness 值。

（7）使用 killall 结束所有 vi 进程。

实训目的： 掌握调整优先级的方法，掌握向进程发送信号的方法。

实训内容：

执行 vi & 命令：

在终端中输入 vi & 命令，启动 vi 编辑器，并将其置于后台运行。

使用 ps 命令查看 vi 进程的 PID：

输入 ps aux | grep vi 命令，查看所有 vi 进程的详细信息，找到您启动的 vi 进程的 PID。

使用 kill 命令结束 vi 进程：

输入 kill -9 PID 命令，其中 PID 是您找到的 vi 进程的 PID，将结束该进程。

使用 nice vi & 启动进程，观察其 niceness 值：

输入 nice vi & 命令，启动 vi 编辑器，并将其置于后台运行，同时观察其 niceness 值。

默认情况下，进程的 niceness 值为 0，表示普通优先级。

使用 nice vi & 并添加合适的参数，使得该进程 niceness 值为负数：

输入 nice -n -5 vi & 命令，启动 vi 编辑器，并将其置于后台运行，同时设置其 niceness 值为 -5，表示高优先级。

负数 niceness 值表示进程具有较高的优先级，可以获得更多的 CPU 时间。

使用 ps 观察前述 vi 进程的 PID 和 niceness：

输入 ps aux | grep vi 命令，查看所有 vi 进程的详细信息，找到您启动的 vi 进程的 PID 和 niceness 值。

使用 renice 分别调大和调小现有的 vi 进程的 niceness 值：

输入 renice 10 -p PID 命令，其中 PID 是您找到的 vi 进程的 PID，将调大该进程的 niceness 值为 10，表示低优先级。

输入 renice -5 -p PID 命令，其中 PID 是您找到的 vi 进程的 PID，将调小该进程的 niceness 值为 -5，表示高优先级。

使用 killall 结束所有 vi 进程：

输入 killall vi 命令，将结束所有 vi 进程。

注意事项：

调整进程的优先级时，需要使用 sudo 命令获取管理员权限。

niceness 值的范围为 -20 到 19，其中 -20 表示最高优先级，19 表示最低优先级。

nice 命令和 renice 命令可以用来调整进程的优先级。

kill 命令和 killall 命令可以用来结束进程。

【实训内容】

（1）执行 sudo apt update 更新系统缓存。

（2）执行 sudo apt list openssh-server 查看软件包信息。

（3）执行 sudo apt install openssh-server 安装 ssh 服务器。

（4）执行 sudo apt show openssh-server 查看软件包相关信息

（5）执行 ssh localhost 测试是否安装成功。

（6）（可选）使用 putty 等工具连接 ssh 服务器。

（7）执行 sudo apt remove openssh-server 卸载该软件包，但会保留该软件包的配置文

档。

（8）执行 sudo apt purge openssh-server 命令，卸载该软件包同时删除配置文件，

（9）如果需要更彻底的删除，可执行 sudo apt autoremove 删除该软件包及其所依赖的、

不再使用的软件包。

实训目的： 熟悉 apt 命令的使用。

实训内容：

更新系统缓存：

输入 sudo apt update 命令，更新系统软件包缓存，以便获取最新的软件包信息。

查看软件包信息：

输入 sudo apt list openssh-server 命令，查看 openssh-server 软件包的安装状态和版本信息。

安装 SSH 服务器：

输入 sudo apt install openssh-server 命令，安装 openssh-server 软件包，这将安装 SSH 服务器。

查看软件包相关信息：

输入 sudo apt show openssh-server 命令，查看 openssh-server 软件包的详细信息，包括描述、版本、依赖关系等。

测试 SSH 服务器：

输入 ssh localhost 命令，尝试连接到本地的 SSH 服务器。

如果您设置了 SSH 密码，则需要输入密码才能连接。

使用 Putty 等工具连接 SSH 服务器 (可选)：

如果您使用 Windows 操作系统，可以使用 Putty 等工具连接到 SSH 服务器。

打开 Putty，输入您的服务器 IP 地址或主机名，选择 SSH 连接方式，然后输入用户名和密码即可连接。

卸载 SSH 服务器：

输入 sudo apt remove openssh-server 命令，卸载 openssh-server 软件包，但会保留其配置文件。

卸载 SSH 服务器并删除配置文件：

输入 sudo apt purge openssh-server 命令，卸载 openssh-server 软件包，并删除其配置文件。

删除 SSH 服务器及其依赖的不再使用的软件包：

输入 sudo apt autoremove 命令，删除 openssh-server 软件包及其依赖的、不再使用的软件包。

注意事项：

在执行 apt 命令时，需要使用 sudo 命令获取管理员权限。

可以使用 apt 命令安装、卸载、更新和查询软件包。

可以使用 apt-cache 命令查询软件包的详细信息。

可以使用 dpkg 命令管理软件包。

【实训内容】

（1）通过百度查找并安装 WPS for linux。

（2）通过百度查找并安装 QQ for linux。

实训目的： 熟悉软件包的查找和安装。

实训内容：

安装 WPS for Linux：

方法一： 打开 Ubuntu 软件中心，搜索 WPS，找到 WPS Office 软件包，点击安装即可。

方法二： 打开终端，输入 sudo apt install wps-office 命令，安装 WPS Office 软件包。

安装 QQ for Linux：

方法一： 打开 Ubuntu 软件中心，搜索 QQ，找到 TIM 软件包，点击安装即可。

方法二： 打开终端，输入 sudo apt install TIM 命令，安装 TIM 软件包。

注意事项：

软件包的名称可能因版本和发行版而异，请根据实际情况选择合适的软件包。

可以使用 dpkg -l | grep 软件包名称 命令查看已安装的软件包列表。

可以使用 dpkg -r 软件包名称 命令卸载软件包。

【实训内容】

（1）将源代码包文件复制到用户主目录中对其解压缩。

（2）阅读其中的 INSTALL 和 README 文件，了解安装事项。

（3）在 Ubuntu 系统上执行 sudo apt install libexpat1-dev 命令安装 expat 库。

（4）执行 sudo apt install libxml2-dev 命令安装 libxml2-dev 包。

（5）下载 APR 源代码包（apr-1.6.5.tar.gz）和 APR-util 源代码包（apr-util-1.6.1.tar.gz），

并进行安装。

（6）到官网 pcre.org 上下载 PCRE 的源代码包（pcre-8.42..tar.gz）并进行安装。

（7）切换到 http-2.4.38 目录，执行 configure 脚本。

（8）运行 make 命令，完成源代码编译。

（9）运行 sudo make install 命令，完成安装。

（ 10 ） 进 行 测 试 。 修 改 Apache 配 置 文 件 /usr/local/apache2/conf/httpd.conf ， 取 消

ServerName 选项的注释。然后切换到/usr/local/apache2/bin 目录，执行 sudo ./apachectl start

启动 Apache。最后使用浏览器访问进行实测。

实训目的： 熟悉源代码安装的操作步骤。

实训准备： 下载 Apache 源代码包 (httpd-version.tar.gz)。

实训步骤：

解压缩源代码包：

将下载的 Apache 源代码包复制到用户主目录中。

在终端中输入 tar -zxvf httpd-version.tar.gz 命令，解压缩源代码包。

阅读 INSTALL 和 README 文件：

进入解压缩后的目录，阅读 INSTALL 和 README 文件，了解安装 Apache 的步骤和注意事项。

安装依赖库：

输入 sudo apt install libexpat1-dev 命令，安装 expat 库。

输入 sudo apt install libxml2-dev 命令，安装 libxml2-dev 包。

安装 APR 和 APR-util：

下载 APR 源代码包 (apr-1.6.5.tar.gz) 和 APR-util 源代码包 (apr-util-1.6.1.tar.gz)。

解压缩这两个源代码包。

进入每个解压缩后的目录，分别执行 ./configure、make 和 sudo make install 命令进行安装。

安装 PCRE：

访问 PCRE 官网 (pcre.org)，下载 PCRE 源代码包 (pcre-8.42.tar.gz)。

解压缩源代码包。

进入解压缩后的目录，执行 ./configure、make 和 sudo make install 命令进行安装。

配置和编译 Apache：

切换到 Apache 源代码目录。

执行 ./configure 命令，配置 Apache 的编译选项。

执行 make 命令，编译 Apache 源代码。

执行 sudo make install 命令，安装 Apache。

测试 Apache：

修改 Apache 配置文件 /usr/local/apache2/conf/httpd.conf，取消 ServerName 选项的注释。

切换到 /usr/local/apache2/bin 目录。

执行 sudo ./apachectl start 命令，启动 Apache 服务器。

打开浏览器，输入 http://localhost 或 http://127.0.0.1，查看 Apache 的测试页面。

注意事项：

在执行 make 和 make install 命令时，需要使用 sudo 命令获取管理员权限。

可以使用 make clean 命令清除编译过程中产生的临时文件。

可以使用 make distclean 命令清除所有编译过程中产生的文件。

安装过程中可能需要根据实际情况调整编译选项。

【实训内容】

（1）显示当前日期时间、执行路径、用户账户及所在的目录位置。参考源码如下：

#!/bin/bash

#这是一个测试脚本

echo –n "当前日期和时间："

date

echo –n "程序执行路径："$PATH

echo "当前登录用户名：`whoami`"

echo –n "当前目录:"

pwd

#end

（2）判断一个文件是不是字符设备文件，并给出相应的提示信息。参考源码如下：

#!/bin/bash

file="/dev/sdb1"

if [ -c $file ]

then

echo "这是字符设备文件"

else

echo "这不是字符设备文件"

fi

（3）从键盘输入两个字符串，比较两个字符串是否相等。参考源码如下：

#!/bin/bash

read -p "请输入两个字符串: " s1 s2

if [s1=s2]

then

11

echo "两个字符串相等"

else

echo "两个字符串不相等"

fi

（4）分别用 for、while 与 until 语句求从整数 1 到 100 的和。

for 语句参考源码：

#!/bin/bash

total=0

for $num in {1..100}

do

total=`expr $total + $num`

done

echo "结果等于： $total"

或者：

#!/bin/bash

total=0

for(($num=1; $num <=100; $num++))

do

total=`expr $total + $num`

done

echo "结果等于： $total"

while 语句参考源码：

#!/bin/bash

total=0

num=0

while [ $num -le 100 ]

do

total=`expr $total + $num`

num=`expr $num + 1`

done

12

echo "结果等于： $total"

until 语句参考源码：

#!/bin/bash

total=0

num=0

until [ $num -gt 100 ]

do

total=`expr $total + $num`

num=`expr $num + 1`

done

echo "计算结果为： $total"

（5）实现每天将主目录下的所有目录和文件归档并压缩为文件 mybackup.tar.gz。

备份文件 backup.sh 参考源码（可使用 nano 编辑器）：

#!/bin/bash

cd /home

tar -zcvf mybackup.tar.gz

将该脚本文件复制到/etc/cron.daily 目录中。

为该脚本文件添加执行权限。

实训目的： 熟悉 Shell 脚本语法，掌握常用程序的编写。

实训内容：

显示当前日期时间、执行路径、用户账户及所在的目录位置：

bash

复制

#!/bin/bash

#这是一个测试脚本

echo -n "当前日期和时间："

date

echo -n "程序执行路径："$PATH

echo "当前登录用户名：`whoami`"

echo -n "当前目录："

pwd

#end

判断一个文件是不是字符设备文件，并给出相应的提示信息：

bash

复制

#!/bin/bash

file="/dev/sdb1"

if [ -c $file ]

then

echo "这是字符设备文件"

else

echo "这不是字符设备文件"

fi

从键盘输入两个字符串，比较两个字符串是否相等：

bash

复制

#!/bin/bash

read -p "请输入两个字符串： " s1 s2

if [ "$s1" = "$s2" ]

then

echo "两个字符串相等"

else

echo "两个字符串不相等"

fi

分别用 for、while 与 until 语句求从整数 1 到 100 的和：

for 循环：

bash

复制

#!/bin/bash

total=0

for num in {1..100}

do

total=$((total + num))

done

echo "结果等于： $total"

while 循环：

bash

复制

#!/bin/bash

total=0

num=0

while [ $num -le 100 ]

do

total=$((total + num))

num=$((num + 1))

done

echo "结果等于： $total"

until 循环：

bash

复制

#!/bin/bash

total=0

num=0

until [ $num -gt 100 ]

do

total=$((total + num))

num=$((num + 1))

done

echo "计算结果为： $total"

实现每天将主目录下的所有目录和文件归档并压缩为文件 mybackup.tar.gz：

创建脚本文件 backup.sh：

bash

复制

#!/bin/bash

cd /home

tar -zcvf mybackup.tar.gz \*

将脚本文件复制到 /etc/cron.daily 目录：

bash

复制

sudo cp backup.sh /etc/cron.daily/

为脚本文件添加执行权限：

bash

复制

sudo chmod +x /etc/cron.daily/backup.sh

注意事项：

在编写 Shell 脚本时，需要使用 #!/bin/bash 声明脚本解释器。

可以使用 echo 命令输出信息。

可以使用 read 命令读取用户输入。

可以使用条件语句 (if、else、elif) 进行条件判断。

可以使用循环语句 (for、while、until) 进行循环操作。

可以使用管道 (|) 和重定向 (>、>>) 进行数据传输。

可以使用变量存储数据。

可以使用数组存储多个数据。

可以使用函数封装代码。

可以使用注释解释代码。

【实训内容】

1、安装配置方案 1：Miniconda 版

（1）安装 Miniconda。

（2）测试 conda 安装是否成功。

（3）conda 源的配置。

（4）安装 Scikit-learn。

（5）测试安装 Scikit-learn 是否成功。

2、安装配置方案 2：Anaconda 版

（1）安装 Anaconda。

（2）测试 conda 安装是否成功。（可跳过）

（3）conda 源的配置。（可跳过）

（4）测试安装 Scikit-learn 是否成功。

3、机器学习应用实例测试。

以教材“13.3.3 实例详解”为基础，进行如下修改。建议初学者每次只进行一项修改。

并观察修改前后的变化。

（1）修改下面语句中的三个坐标值中的一个或者多个。

centers = [[-1, 1], [1, 1], [1, -1]]

（2）修改 cluster\_std 值，可以调大和调小。

X, \_ = make\_blobs(n\_samples=10000, centers=centers, cluster\_std=0.6)

（3）删除引号中字符串的前 1 位或者前几位。

colors = cycle('grcmykbgrcmykbgrcmykbgrcmykb')

实训目的： 掌握配置机器学习开发环境的基本方法。

实训内容：

1. 安装配置方案 1：Miniconda 版

(1) 安装 Miniconda

访问 Miniconda 官网 (https://docs.conda.io/en/latest/miniconda.html)，下载适合您操作系统的 Miniconda 安装包。

运行安装包，按照提示完成安装。

(2) 测试 conda 安装是否成功

在终端中输入 conda --version 命令，查看 conda 的版本信息。

(3) conda 源的配置

可以配置 conda 源，以便更方便地安装软件包。

常用的 conda 源包括：

https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/

https://mirrors.aliyun.com/anaconda/pkgs/main/

https://mirrors.ustc.edu.cn/anaconda/pkgs/main/

(4) 安装 Scikit-learn

在终端中输入 conda install scikit-learn 命令，安装 Scikit-learn。

(5) 测试安装 Scikit-learn 是否成功

在终端中输入 python 命令进入 Python 交互模式。

输入 import sklearn 命令，测试是否成功导入 Scikit-learn 库。

2. 安装配置方案 2：Anaconda 版

(1) 安装 Anaconda

访问 Anaconda 官网 (https://www.anaconda.com/products/distribution)，下载适合您操作系统的 Anaconda 安装包。

运行安装包，按照提示完成安装。

(2) 测试 conda 安装是否成功

测试 conda 安装是否成功可以通过以下几种方法：

方法一：查看 conda 版本信息

在终端中输入 conda --version 命令。

如果 conda 安装成功，则会显示 conda 的版本信息，例如：

复制

conda 4.10.3

方法二：查看 conda 环境列表

在终端中输入 conda env list 命令。

如果 conda 安装成功，则会显示当前用户可用的 conda 环境列表，例如：

复制

# conda environments:

#

base \* /Users/yourusername/miniconda3

方法三：创建 conda 环境

在终端中输入 conda create -n myenv python=3.8 命令，创建一个名为 myenv 的 conda 环境，指定 Python 版本为 3.8。

如果 conda 安装成功，则会创建环境并显示相关信息，例如：

复制

Collecting package metadata (current\_repodata.json): done

Solving environment: done

## Package Plan ##

environment location: /Users/yourusername/miniconda3/envs/myenv

The following NEW packages will be INSTALLED:

ca-certificates pkgs/main/noarch::ca-certificates-2021.10.26-h033912b\_2

certifi pkgs/main/noarch::certifi-2021.10.8-pyhd3deb0d\_2

openssl pkgs/main/osx-64::openssl-1.1.1k-h1de35cc\_0

pip pkgs/main/noarch::pip-21.2.4-pyhd3deb0d\_0

python pkgs/main/osx-64::python-3.8.12-h5d3a9e2\_2

python\_abi pkgs/main/noarch::python\_abi-3.8-2\_cp38

readline pkgs/main/osx-64::readline-8.1-h1de35cc\_0

setuptools pkgs/main/noarch::setuptools-58.0.4-py38h0b31af3\_0

sqlite pkgs/main/osx-64::sqlite-3.36.0-h5f5109e\_0

tk pkgs/main/osx-64::tk-8.6.10-h1de35cc\_0

wheel pkgs/main/noarch::wheel-0.37.0-pyhd3deb0d\_0

zlib pkgs/main/osx-64::zlib-1.2.11-h1de35cc\_3

方法四：安装 Python 包

在终端中输入 conda install numpy 命令，安装 NumPy 包。

如果 conda 安装成功，则会安装 NumPy 并显示相关信息，例如：

复制

Collecting package metadata (current\_repodata.json): done

Solving environment: done

## Package Plan ##

environment location: /Users/yourusername/miniconda3

The following packages will be downloaded:

package | build

---------------------------|-----------------

numpy-1.21.2 | pypi\_0 20.6 MB pypi

------------------------------------------------------------

Total: 20.6 MB

The following NEW packages will be INSTALLED:

numpy pkgs/main/noarch::numpy-1.21.2-pyhd3eb1b0\_0

注意事项：

如果以上任何方法无法成功执行，则说明 conda 安装可能存在问题。

可以参考 conda 官方文档 (https://docs.conda.io/en/latest/) 获取更多信息。

(3) conda 源的配置

配置 conda 源可以方便您更快地下载和安装软件包。以下是配置 conda 源的步骤：

方法一：使用 conda config 命令

在终端中输入以下命令：

复制

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/r/

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/pro/

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/msys2/

以上命令将添加清华大学开源软件镜像站作为 conda 源。

您可以根据需要添加其他 conda 源，例如：

复制

conda config --add channels https://mirrors.aliyun.com/anaconda/pkgs/main/

conda config --add channels https://mirrors.ustc.edu.cn/anaconda/pkgs/main/

方法二：修改 .condarc 文件

打开终端，输入 cd ~ 命令进入用户主目录。

使用文本编辑器打开 .condarc 文件，例如：

复制

nano .condarc

在 .condarc 文件中添加以下内容：

复制

channels:

- defaults

- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/

- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/

- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/r/

- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/pro/

- https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/msys2/

保存并关闭 .condarc 文件。

注意事项：

defaults 是 conda 默认的软件源，建议保留。

可以添加多个 conda 源，按照优先级从高到低排列。

更换 conda 源后，需要执行 conda clean -i 命令清除旧的缓存，才能生效。

常用的 conda 源：

清华大学开源软件镜像站： https://mirrors.tuna.tsinghua.edu.cn/anaconda/

阿里云开源镜像站： https://mirrors.aliyun.com/anaconda/

中国科学技术大学开源镜像站： https://mirrors.ustc.edu.cn/anaconda/

通过配置 conda 源，您可以更方便地下载和安装软件包，提高工作效率。

(4) 测试安装 Scikit-learn 是否成功

与 Miniconda 版本相同，在 Python 交互模式中测试是否成功导入 Scikit-learn 库。

3. 机器学习应用实例测试

以教材“13.3.3 实例详解”为基础，进行如下修改：

修改坐标值： 修改 centers 变量的值，例如 centers = [[-2, 2], [2, 2], [2, -2]]。

修改 cluster\_std 值： 修改 X, \_ = make\_blobs(n\_samples=10000, centers=centers, cluster\_std=0.8) 中的 cluster\_std 值，例如 0.8。

修改颜色： 修改 colors = cycle('grcmykbgrcmykbgrcmykbgrcmykb') 中的字符串，例如 colors = cycle('rbgkmyc')。

注意事项：

Miniconda 和 Anaconda 都是 Python 的科学计算和数据分析平台，都包含 conda 包管理器。

Scikit-learn 是 Python 的机器学习库，用于数据挖掘和数据分析。

可以使用 Jupyter Notebook 等工具编写和运行机器学习代码。

可以使用 Scikit-learn 的各种算法进行机器学习任务，例如分类、回归、聚类等。

【实训内容】

1、安装 TensorFlow。测试是否安装成功

2、深度学习应用实例。

以教材“13. 5 实例详解”为基础，进行如下修改。建议初学者每次只进行一项修改。

并观察修改前后的变化。

（1）将函数表达式 y = x + 3 修改为 y =2-3\*x。主要涉及下面两条语句。

y = x + 3 + np.random.randn(100) \* 0.2

plt.plot(x, x +3, color = 'g' ,linewidth=2)

plt.plot(x,x + 3,label="Reference curve",color="r",linewidth = 2)

（2）修改噪声值，观察训练效果。可以分别将噪声幅值 0.2 这个值调大和调小，观察

拟合效果。

y = x + 3 + np.random.randn(100) \* 0.2

实训目的： 掌握配置深度学习开发环境的基本方法。

实训内容：

1. 安装 TensorFlow

在终端中输入 conda install tensorflow 命令，安装 TensorFlow。

可以选择安装 TensorFlow 的 CPU 版本或 GPU 版本。

安装完成后，输入 python 命令进入 Python 交互模式，输入 import tensorflow as tf 命令测试是否成功导入 TensorFlow 库。

2. 深度学习应用实例

以教材“13.5 实例详解”为基础，进行如下修改：

(1) 修改函数表达式

将 y = x + 3 + np.random.randn(100) \* 0.2 修改为 y = 2 - 3 \* x + np.random.randn(100) \* 0.2。

修改 plt.plot(x, x + 3, color = 'g' ,linewidth=2) 为 plt.plot(x, 2 - 3 \* x, color = 'g' ,linewidth=2)。

修改 plt.plot(x,x + 3,label="Reference curve",color="r",linewidth = 2) 为 plt.plot(x, 2 - 3 \* x,label="Reference curve",color="r",linewidth = 2)。

修改后的代码示例：

python

复制

import numpy as np

import matplotlib.pyplot as plt

import tensorflow as tf

# 生成数据

x = np.linspace(-1, 1, 100)

y = 2 - 3 \* x + np.random.randn(100) \* 0.2

# 创建模型

model = tf.keras.Sequential([

tf.keras.layers.Dense(1, input\_shape=[1])

])

# 编译模型

model.compile(optimizer='sgd', loss='mse')

# 训练模型

model.fit(x, y, epochs=100)

# 测试模型

predictions = model.predict(x)

# 绘制结果

plt.scatter(x, y, color='b')

plt.plot(x, predictions, color='r')

plt.plot(x, 2 - 3 \* x, color='g', linestyle='--')

plt.legend(['Data', 'Predictions', 'Reference'])

plt.show()

(2) 修改噪声值

将 y = x + 3 + np.random.randn(100) \* 0.2 中的 0.2 修改为其他值，例如 0.1 或 0.3。

观察噪声幅值的变化对模型训练效果的影响。

注意事项：

TensorFlow 是一个开源的深度学习框架，用于构建和训练深度学习模型。

可以使用 TensorFlow 的各种 API 进行深度学习开发。

可以使用 TensorFlow 的可视化工具 TensorBoard 查看模型训练过程。

通过本实训，您可以学习如何配置深度学习开发环境，并使用 TensorFlow 进行简单的深度学习任务。