

# Open Data/Open Source: Seismic Unix scripts to process a 2D land line

**Karl Schleicher**

*k\_schleicher@hotmail.com*

*John A. and Katherine G. Jackson School of Geosciences*

*The University of Texas at Austin*

*University Station, Box X*

*Austin, TX 78713-8924*

(January 29, 2012)

Running head: **Alaska**

## **ABSTRACT**

This paper describes how to process an internet downloadable 2D land line data set through a very basic processing sequence using Seismic Unix. The data from the Alaska North Slope has good signal, although it may be suitable for testing ground-roll and noise burst attenuation programs. The detailed steps to download the dataset from the Internet and process it are described. You should be able to follow these steps and recreate my results. I hope this will accelerate the testing and validation of programs developed in different research groups.

The paper starts with background information such as data downloading, region overview, data acquisition, and previous processing. The scripts and custom programs for Seismic Unix translation, header load, qc, shotrecord velocity filter, deconvolution, CDP gather, velocity analysis, residual statics, stack, and post stack migration can be downloaded to

your computer.

Header loading is not well described in other Seismic Unix documentation or papers. One approach to header loading is described here. The velocity analysis script is more advanced than what I found in previous publications.

This paper is the beginning of an effort to build an open data/open source library for research, software testing, demonstrations, and user training. I encourage you to use these scripts and improve on my results. You can send me email for the latest results on this line, and progress on other software and data.

## INTRODUCTION

Seismic Unix (SU) is a popular open-source seismic processing software package (Stockwell, 1997; Cohen and Stockwell, 2010). The distribution includes some example processing scripts and three books that teach how to use the software are *Seismic Processing with Seismic Unix* (Forel et al., 2005), *Geophysical Image Processing with Seismic Unix* (Stockwell, 2011), and *Theory of Seismic Imaging* (Scales, 1997) . These reference cover broad subjects including basic Unix commands, setting up a user environment, basic seismic processing, migration theory, programming, and advanced scripting for seismic processing. You can follow the text and recreate some seismic processing examples. These books do not provide good examples of geometry definition and how to load trace headers.

I hope to assist a processing geophysicst working or studing independently at a University or company that needs field data. I assume he is familiar with Seismic Unix. I providing example scripts, programs, and field data to process a 2D land line using the processing system. I describe the data acquisition and previous data processing. I provide the custom programs that load the trace headers. The scripts for data reformat to Seismic Unix format, header loading qc display, shot record velocity filter, deconvoution, CDP gather, velocity analysis, residual statics, stack, and post stack migration can be downloaded and this paper describes the scripts. This is a more complete processing sequence and a different dataset than provided in previous publications. The scripts provide the detailed parameters used by the Seismic Unix programs. The parameters for deconvolution, velocity filter, mute, moveout velocity, and scaling can adapted for use by other seismic processing systems. The results are similar to the results obtained in 1981 using GSI's proprietary software.

You can download Seismic Unix, the field data, and the scripts, and recreate my results.

The scripts, data, and programs can be modified to validate your own seismic processing ideas. I intend the reader to use the paper and the scripts together to jump directly to a processing stage of interest to change parameters or programs to test alternative processing. This should accelerate the testing and validation of new seismic research, especially at universities and small companies that do not have large data archives and processing groups.

## BACKGROUND INFORMATION ABOUT THE DATA

President Warren Harding created the Naval Petroleum Reserve Number 4 in 1923. It was renamed the National Petroleum Reserve, Alaska (NPRA) in 1976. Figure 1 is a map of the region and Figure 2 maps the lines collected and processed between 1974 and 1981. I selected and processed line 31-81 because it was a short line from the last acquisition season.

## SETTING UP AND READING THE DATA DOCUMENTATION

Download Madagascar and you should have all files to process line 31-81 in the directory:  
`$RSFSRC/book/data/alaska/line31-81.`

Change into this directory. This paper lists commands and the `SConstruct` file contains the rules that translate the commands into SU tasks to process the data.

An alternative to installing Madagascar is to go to the website [http://dl.dropbox.com/u/37269048/open\\_lib\\_demo\\_2011.tar.gz](http://dl.dropbox.com/u/37269048/open_lib_demo_2011.tar.gz)

This will download a tar file that you can install. On my system I do this with the commands:



```
cd  
  
mv Downloads/rsf-alaska.tar.gz .  
  
tar xvzf rsf-alaska.tar.gz
```

There used to be a higher resolution image of the stack display, the file S609.TIF. This file is no longer available on the USGS Internet site. I was able to view this file in OpenOffice.org draw and zoom to read the side label that describes the acquisition and processing parameters.

The data was acquired 96 trace, 12 fold with a dynamite source. The shotpoint interval is 440 ft and the receiver interval is 110 ft. The average shot depth is 75 ft. The elevation plot on the section header shows the elevation is about 200 ft and there is little variation.

The processing applied by GSI in 1981 was:

1. Edit and true amplitude recovery
2. Spherical divergence and exponentiation (alpha - 4.5 db/s, initial time - 0 s, final time - 4.5 s)
3. Velocity filtering dip range -12 to 4 ms/trace
4. Designature deconvolution
5. Time-variant scaling (unity)
6. Apply datum statics
7. Velocity estimation
8. Normal moveout correction

9. Automatic Residual Statics application
10. First Break Suppression
  - 100 ms at 55 ft
  - 380 ms at 2970 ft
  - 700 ms at 5225 ft
11. Common depth point stack
12. Time Variant filtering
  - 16-50 Hz at 700 ms
  - 14-45 Hz at 1400 ms
  - 12-40 Hz at 2000 ms
  - 8-40 Hz at 400 ms
13. Time variant scaling

The surveyor and observer logs are downloaded and displayed using the commands:

```
scons surveylog.view  
scons observerlog.view
```

This can be done using the data on the tar file by typing:

```
./surveylog.job  
./observerlog.job
```

I read the surveyor's log and typed the elevations into the `spnElev.txt` file. The `spnElev.txt` file contains two columns. The first is the shotpoint number and the second

is the elevation. I read the observer's log and typed the relationship between the field record numbers the shotpoint numbers into the `recnoSpn.txt` file. The file `recnoSpn.txt` contains two columns. The first column is the field record number and the second column is the shotpoint number. I used a shotpoint number -999999 to reject bad field records. These files are used in the next section to load the trace headers.

## DATA LOADING AND INITIAL DATA QC

I found little documentation about loading data and creating trace headers in previous publications. This section described how I loaded the headers.

The three field data SEGY files are downloaded, converted to Seismic Unix format, concatenated, and a print created of the range of each trace header using the command:

```
scons list1.su
```

Or to use the tar file type:

```
./list1.job
```

The resulting listing is:

6868 traces:

```
trac1    1 6868 (1 - 6868)
tracr    1 6868 (1 - 6868)
fldr     101 168 (101 - 168)
tracf    1 101 (1 - 101)
trid      1
```

```
ns      3000
dt      2000
f1      0.000000 512.000000 (0.000000 - 512.000000)
```

The print indicates there is little information in the input headers. There are 31 shot records (`fldr` 101 - 130) and each record has 101 traces (`tracf` 1 - 101). The trace headers will be created using `fldr` (the field record number) and `tracf` (the field trace number). A list of these headers on the first 3000 traces is created using:

```
scons list2.su
```

If you are using the tar file type:

```
./list2.job
```

Part of the print is:

```
fldr=101 tracf=1  cdp=0  cdpt=0  offset=0
fldr=101 tracf=2  cdp=0  cdpt=0  offset=0
fldr=101 tracf=3  cdp=0  cdpt=0  offset=0
...
fldr=101 tracf=101 cdp=0  cdpt=0  offset=0
fldr=102 tracf=1   cdp=0  cdpt=0  offset=0
fldr=102 tracf=2   cdp=0  cdpt=0  offset=0
...
```

A first display of the seismic data and a zoom of the same data (figures 3 and 4) are produced with the commands:

```
scons first.view
```

```
scons zoomfirst.view
```

If you are using the tar file type:

```
./first.job
```

```
./zoomfirst.job
```

The first 10 records in the file are test records. These test records are the unusual traces in the left third of Figure 3. The Observer Log does not mention these test records, but indicates the first shotpoint is field file identifier (ffid) 110. There are also 5 auxiliary traces in each shot record. These auxiliary traces can be seen on Figure 4.

The display of shotpoint 24 (Figure 4) is displayed with the command:

```
scons firstrec24.view
```

If you are using the tar file type:

```
./firstrec24.job
```

This display as created using suwind to select traces 2324 through 2424, so it is the 24th record in the input file (including the test records). This plot confirms that the 5 auxiliary traces are `tracf 97` through 101.

The movie display is not included in this paper. The movie loops continuously and can be stopped by pressing 's'. Once stopped, you can move forward or backward using the 'f' and 'b' keys. The movie can be run with the command:

```
scons firstmovie.view
```

If you are using the tar file type:

```
./firstmovie.job
```

The trace headers are loaded using the command:

```
scons allshots.su
```

Or from the tar file:

```
./allshots.job
```

The stages in loading the trace header are

1. Run `sugethw` to create the file `hdrfile.txt` containing one record for each trace with  
`trac1, fldr, tracf`
2. Run the custom Python program `InterpText.py`. This program
  - (a) reads `recnoSpn.txt` that defines the `recno/spn` relationship
  - (b) reads `spnElev.txt` the defined the elevation of each shot
  - (c) interpolates this data and output `hdrfile1.txt` with values for all keys to be loaded
3. converts `hdrfile1.txt` to binary format (`binary_hdrfile1.dat`)
4. loads the trace headers
5. removes bad shotpoints and CDPs (with value=-999999)
6. removes the bad shotpoint 149

## 7. outputs `allshots.su`

`InterpText.py` is a custom Python script to load the trace headers. The code is not elegant, but it is straightforward and works. This program requires files to define the shotpoint elevation and the 'ffid'/'ep' relationship. The surveyor log defines the shotpoint elevations. The observers log describes the relationship between the su header keys 'ffid' and 'ep' (these are called 'REC' and 'SHOTPOINT' by in the observers log). This information was typed into two files (`recnoSpn.txt` and `spnElev.txt`). In addition to interpolating the record number/shotpoint table and the elevation data the program also computes the geometry including the receiver locations and the CDP numbers.

The stack file from the previous process is downloaded and displayed using the commands:

```
scons checkstack.view
```

```
scons zoomcheckstack.view
```

If you are using the tar file type:

```
./checkstack.job
```

```
./zoomcheckstack.job
```

The first result is shown in Figure 6. The zoom display is not shown.

## SHOT RECORD VELOCITY FILTER

I applied velocity filtering (`sudipfilt`) to remove the ground roll. I processed the receivers leading the shotpoint separately from the trailing receivers (i.e. I split the shots). I decided

to split the shots so that I could use an asymmetrical dip filter (-15,5 ms/trace). That dip filter is similar to the 1981 processing that used (-12,4). Sudipfilter was intended for post stack processing, so I wrote the script `alaskavelfilt.sh` to loop over the shots and divide individual shotrecords into positive and negative offsets. Figure 7 shows the Figure 4 shotpoint with velocity filtering applied.

The command:

```
scons velfiltrec24.view
```

or using the tar file:

```
../velfiltrec24.job
```

applies velocity filtering and produces the figure.

## CDP SORT AND MUTE

The command

```
scons cdp250-251.view
```

or using the tar file:

```
./cdp250-251.job
```

sorts the data to CDP order, applies a mute, and displays CDPs 250 and 251. This result is shown in Figure 8. I picked the mute from the stacked section. The first trace on the stack is created from a single far offset which has moveout applied. In order to recreate this mute



I applied moveout using a single velocity function contained in the file `vbrureorig.txt`. After applying the mute, I reversed the moveout.

## VELOCITY ANALYSIS AND RESIDUAL STATICS STRATEGY

A problem encountered in residual statics estimation is the coupling of velocity and statics. Changing the medium wavelength statics (0.5 to 3 spread length) will change the medium wavelength stacking velocity. The classic approach to this problem is to repeat the velocity analysis and residual static estimation sequence. Incorporating rapid lateral variation in the initial stacking velocity will prevent the medium wavelength estimation by residual statics. The undesirable rapid velocity variation is locked into the final velocity field. I used the strategy described on page 236 of Mike Cox' statics book (Cox, 1999). I ran velocity analysis and statics twice. On the first iteration, I used single function velocity function for moveout before residual static estimation. If the line was longer I might have run a coarsely sampled set of velocity analyzes. After the first pass of residual statics, I ran velocity analysis every 48 CDPs (2640 ft or 0.5 spreadlength). I did not notice improvement when I compared the stack with the first pass of residual statics and the velocity field from the 48 CDP velocity analysis with the results after two passes of residual statics. The next sections step through the velocity analysis and residual static processes.

## FIRST PASS VELOCITY ANALYSIS

I used a long script, `iva.sh`, that combines several SU programs for velocity interpretation. My `iva.sh` is a combination of the scripts `iva.sh` and `velanQC.sh` in sections 7.6.7.3 and 8.2.2.2 of *Seismic Processing with Seismic Unix* (Forel et al., 2005). My script is more

practical, because you can start from no velocity function, or review and update an existing velocity field. The script has more capabilities, is tidier, and is a little shorter than Forel's script.

Figure 9 is an example velocity analysis plot. It can be produced with the commands:

```
rm vbrute.txt  
scons vbrute.txt
```

Or using the tar file:

```
./vbrute.job
```

There are four components of the plot: semblance plot, constant velocity stack (CVS) plot, cmp gather without NMO, and cmp gather after NMO. The semblance plots and CVS plot have the velocity function over plotted. The velocity analysis can be picked by pointing on the semblance and 'selecting' by pressing the 's' key. To 'quit' the location press the 'q' key. If you have 'selected' any picks, the plots will be recreated with your new velocity function, otherwise processing will continue at the next location. (The brute analysis only has one velocity location, so 'q' just quits the velocity analysis.)

The velocities I picked are in the `vbruteorig.txt` file.

## BRUTE STACK

The brute stack can be produced with the command:

```
scons brutestack.view
```

or from the tar file:

```
./brutestack.job
```

This command reads the cdp gather data and applies decon using supef. In order to define a decon design gate that depends on offset, I used sustatic to time shift the data to line up where I wanted the time gate to start. After decon (supef) the time shift is removed. Moveout using the brute velocity is applied followed by agc and a mute. This data is written to a file which is reread to create the brute stack shown in Figure 10. You can create a zoom of this plot using the command:

```
scons zoombrutestack.view
```

or

```
./zoombrutestack.job
```

You can make a movie of the gathers with moveout applied with the command:

```
scons movie_velfiltcdpsnmo.view
```

or

```
./movie_velfiltcdpsnmo.job
```

## **FIRST PASS RESIDUAL STATICS AND SECOND PASS VELOCITY ANALYSIS**

The command:

```
scons rstack.view
```

or

```
./rstack.job
```

Will create a stack with residual statics and a detailed velocity analysis by the following steps:

1. Read the gathers created in the brute stack section
2. Band pass filter, select a time window, and up sample to 1 ms
3. Compute residual statics
4. Reread the gathers created in the brute stack section
5. Apply the residual statics
6. Remove moveout with the single velocity function
7. Second pass velocity estimation every 48 cdps (2640 ft)
8. Normal moveout correction with the second pass velocity field
9. Stack

The residual statics stack is shown in Figure 11. A zoom plot can be produce with the command:

```
scons zoomrstack.view
```

or

```
./zoomrstack.job
```

The lateral variable stacking velocity field shown in Figure 12 can be created with the command:

```
scons vfile.view
```

or

```
./vfile.job
```

## SECOND PASS RESIDUAL STATICS

The command:

```
scons rstack1.view
```

or

```
./rstack1.job
```

creates a stack with second pass residuals shown in Figure 13. I compared figures 11 and 13 and did not see any improvement.

A zoom plot can be produced with the command:

```
scons zoomrstack1.view
```

or

```
./zoomrstack1.job
```

I enjoyed comparing the zoom plots of the initial brute stack and the stack with two iterations of residual statics and velocity analysis. These plots are not included in the paper, but you can create them with the commands:

```
scons zoombrutestack.view &
```

```
scons zoomrstack1.view &
```

or

```
./zoombrutestack.job &
```

```
./zoomrstack1.job &
```

Make the plots full screen size and toggle between the windows using the buttons on the bottom run bar. The event disruptions that line up vertically are static problems solved by the residual statics process.

## POST STACK MIGRATION

The commands:

```
scons mig.view
```

```
scons migps.view
```

or

```
./mig.job
```

```
./migps.job
```

apply post stack to the second pass residual statics stack (Figure 13). The first command produces the Kirchhoff migration shown in Figure 14 and the second command produces the phase shift migration which is not shown. The Kirchhoff data used the final stacking velocity field (Figure 12). The phase-shift migration used a single interval velocity. I converted to interval velocity using Dix' equation (Dix, 1955) in a spreadsheet.

## COMPARISON OF THE 2011 AND 1981 RESULTS

Figures 6 and 13 are the results by GSI using proprietary software in 1981 and the results obtained using SU. Some of the differences in the processing sequences are:

1. The 1981 processing used GSI's designature process. This source signature removal technique applies the same operator to all traces in a shotrecord. The filter is designed differently from conventional deconvolution and the two types of deconvolution will not produce data with the same phase.
2. There are many differences in the application of AGC. For example GSI applied AGC before velocity filtering and inverse AGC after velocity filter.
3. GSI used diversity stack to control the high amplitude noise bursts.

Considering all these differences, I think the results are surprisingly similar, especially below 400 ms. The shallow results are better on the 1981 processing.

## DISCUSSION OF THE SEISMIC UNIX PROCESSING

I found the SU software hard to use. The `sudipfilter` program was not intended for shotrecord velocity filtering, so a collection of programs were required. Many of my input

errors were not trapped by the code so I struggled to debug my scripts. SU is a useful prototyping environment, but it falls short of the packages I used in industry. I look forward to future improvements.

## KEY DATASETS PRODUCED

This processing sequence produces many datasets, but the best files for additional processing are:

1. allshots.su - unprocessed field data with geometry in headers. Bad traces and aux traces removed.
2. velfiltcdpsmuterstat - preprocessed cdp gathers with residual static applied ready for normal moveout.
3. vpickorig.txt - final stacking velocities.
4. rstack1.su - final residual statics stack.
5. mig.su - post stack kirchhoff migrated data.

## PLANS

Some of the ideas I have to continue this work include:

1. Process using other software (open-source or proprietary).
2. Study difference and see if the open-source software can be improved.
3. Working on different datasets (marine, 3D, ...).



## CONCLUSIONS

I have provided a set of scripts for processing a land line using Seismic Unix. These scripts include the following processing stages:

1. Data load
2. Trace header creation
3. Velocity filtering
4. CDP gather
5. Brute velocity analysis
6. First pass residual statics
7. Final velocity analysis
8. Second pass residual statics
9. Stack
10. Post stack Kirchhoff time migration
11. Post stack phase shift migration

My processing demonstrates processing stages not presented by Forel or Stockwell. The results can be recreated by others since the processing scripts and the data can be downloaded from the Internet.

The task of loading trace headers is not addressed in other publications. The custom Python program included in this paper is one way to accomplish this processing stage. The

velocity analysis script improves the scripts in Forel's book. I was able to pick the velocities on this line using the script.

Some of the ideas I have to continue this work include:

1. Improve results by selecting different SU programs or different parameters.
2. Process using other software (open-source or proprietary).
3. Working on different datasets (marine, 3D, ...).
4. Contribute source improvements back to SU.

## **ACKNOWLEDGMENTS**

I thank National Energy Research Seismic Library (NERSL) for providing download access of this 2D land line from Alaska. I thank the Bureau of Economic Geology at the University of Texas for supporting my Open Data/Open Source effort.

## REFERENCES

- Cohen, J. K., and J. W. Stockwell, 2010, CWP/SU: Seismic Un\*x Release No. 42: an open source software package for seismic research and processing: Center for Wave Phenomena, Colorado School of Mines.
- Cox, M., 1999, Static Corrections for Seismic Reflection Surveys: Soc. of Expl. Geophys.
- Dix, C. H., 1955, Seismic velocities from surface measurements: *Geophysics*, **20**, 68–86.
- Forel, D., T. Benz, and W. D. Pennington, 2005, Seismic data processing with Seismic Un\*x: Society of Exploration Geophysicists.
- Scales, J. A., 1997, Theory of seismic imaging: Samizdat Press, <http://www.samizdat.mines.edu/imaging>.
- Stockwell, J. W., 1997, Free software in education: A case study of CWP/SU: Seismic Unix: *The Leading Edge*, **16**, 1045–1049.
- , 2011, Geophysical image processing with Seismic Unix: Course notes for GPGN461/561: Colorado School of Mines, <http://www.cwp.mines.edu/~john/GPGN461.561>.

## LIST OF FIGURES

1 Map of the National Petroleum Reserve in Alaska from the Wikipedia entry about the National Petroleum Reserve Alaska.

2 Map of seismic lines collects between 1974 and 1981 from the Internet site <http://energy.usgs.gov/GeochemistryGeophysics/SeismicDataProcessingInterpretation/NPRASEismicData.aspx>.

3 First 3000 traces on the first SEG Y file.

4 Zoom of Figure 3. The first look at the data. Notice the ground roll and good signal. There are 101 traces in each shotpoint, 5 auxiliary traces and 96 data traces.

5 The 24th shot record without processing. The last 5 traces are auxiliary traces. There is noise on the traces closest to the shotpoint. One trace has a noise burst near the maximum time.

6 Plot of the SEG Y of the final stack from the 1981 processing.

7 The 24th shot record after agc and velocity filter.

8 CDP gathers 250 and 251 with mute applied.

9 Example velocity analysis from iva.sh.

10 Brute stack.

11 Stack with first pass residual statics and a lateral variable stacking velocity field.

12 The lateral variable stacking velocity field interpreted after residual statics.

13 The final stack with the lateral variable stacking velocity and a second pass of residual statics.

14 Post stack Kirchhoff migration of the final stack.



Figure 1: Map of the National Petroleum Reserve in Alaska from the Wikipedia entry about the National Petroleum Reserve Alaska.

Schleicher –



Figure 2: Map of seismic lines collects between 1974 and 1981 from the Internet site <http://energy.usgs.gov/GeochemistryGeophysics/SeismicDataProcessingInterpretation/NPRASEismicDataArchive.aspx>.

Schleicher –

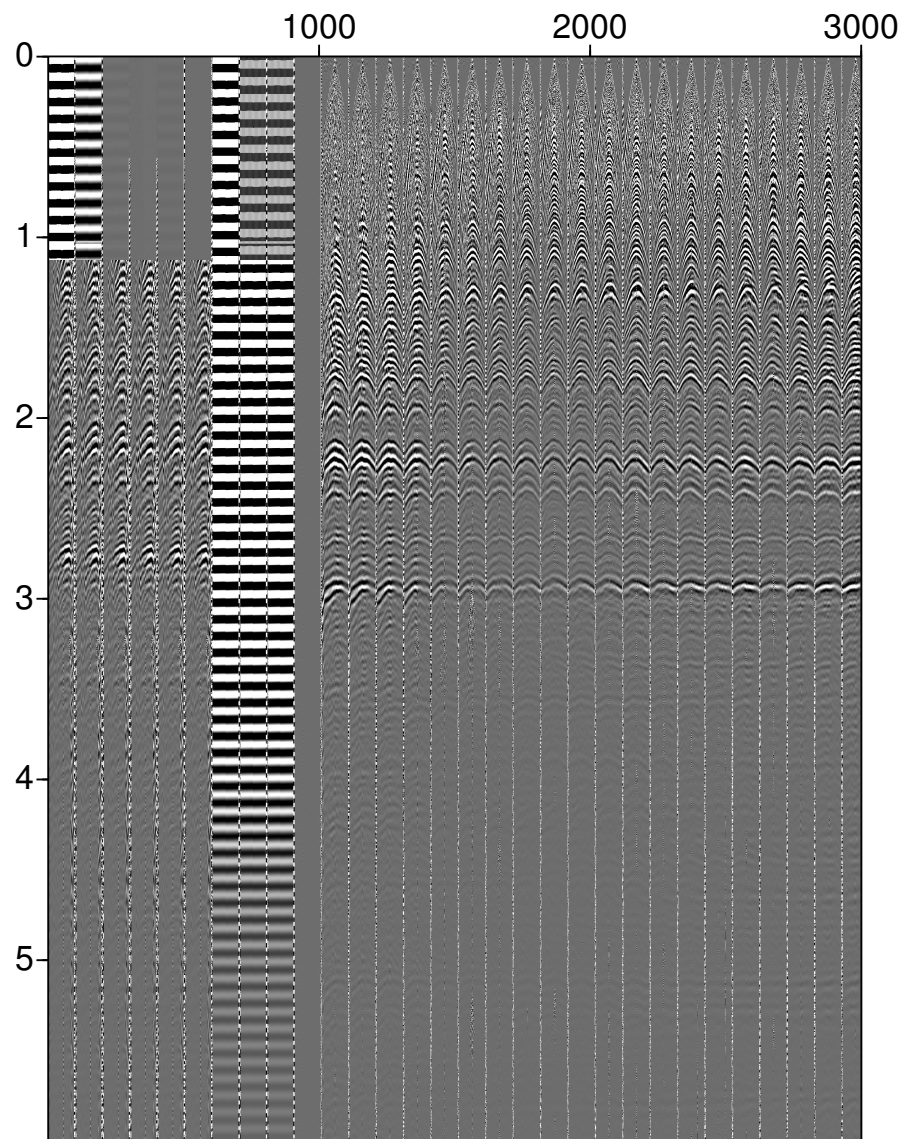


Figure 3: First 3000 traces on the first SEGY file. **Schleicher** –

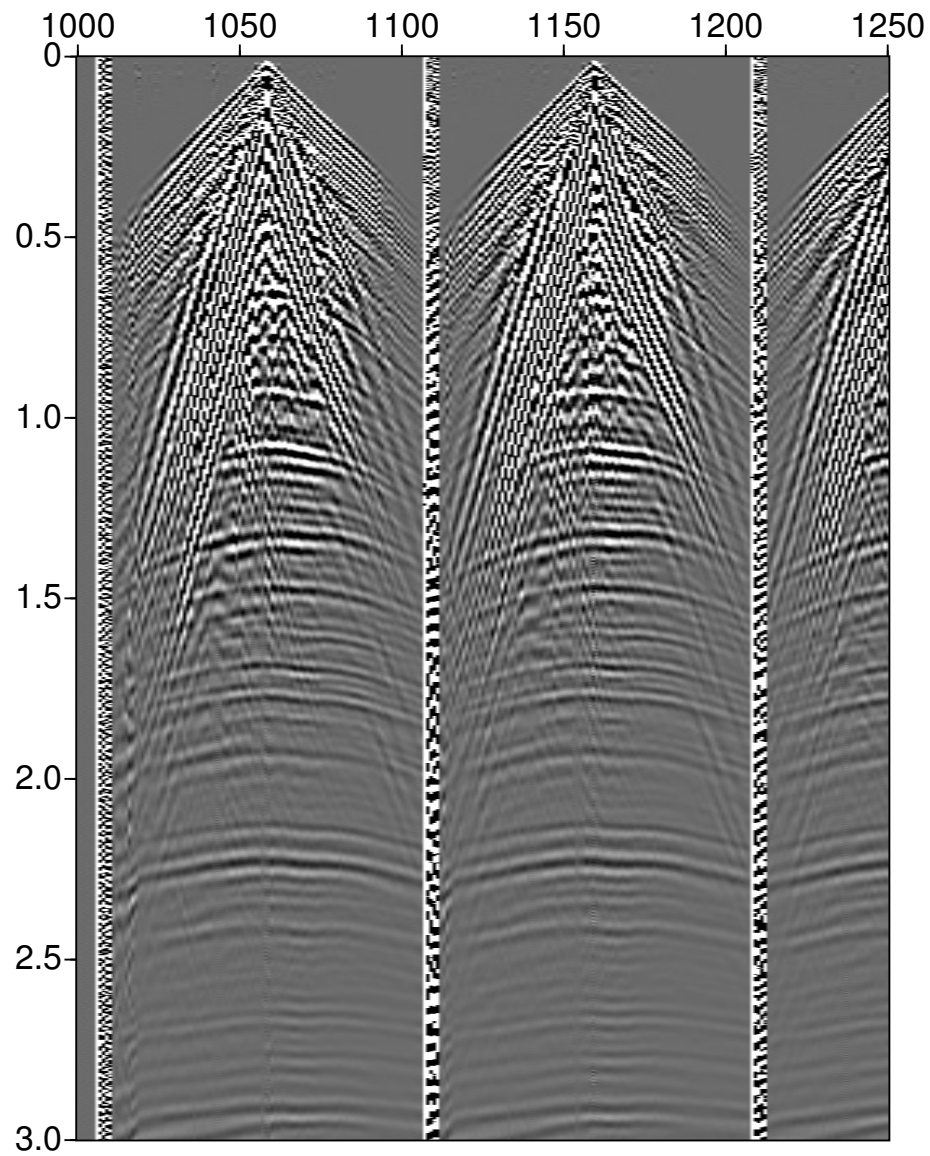


Figure 4: Zoom of Figure 3. The first look at the data. Notice the ground roll and good signal. There are 101 traces in each shotpoint, 5 auxiliary traces and 96 data traces.



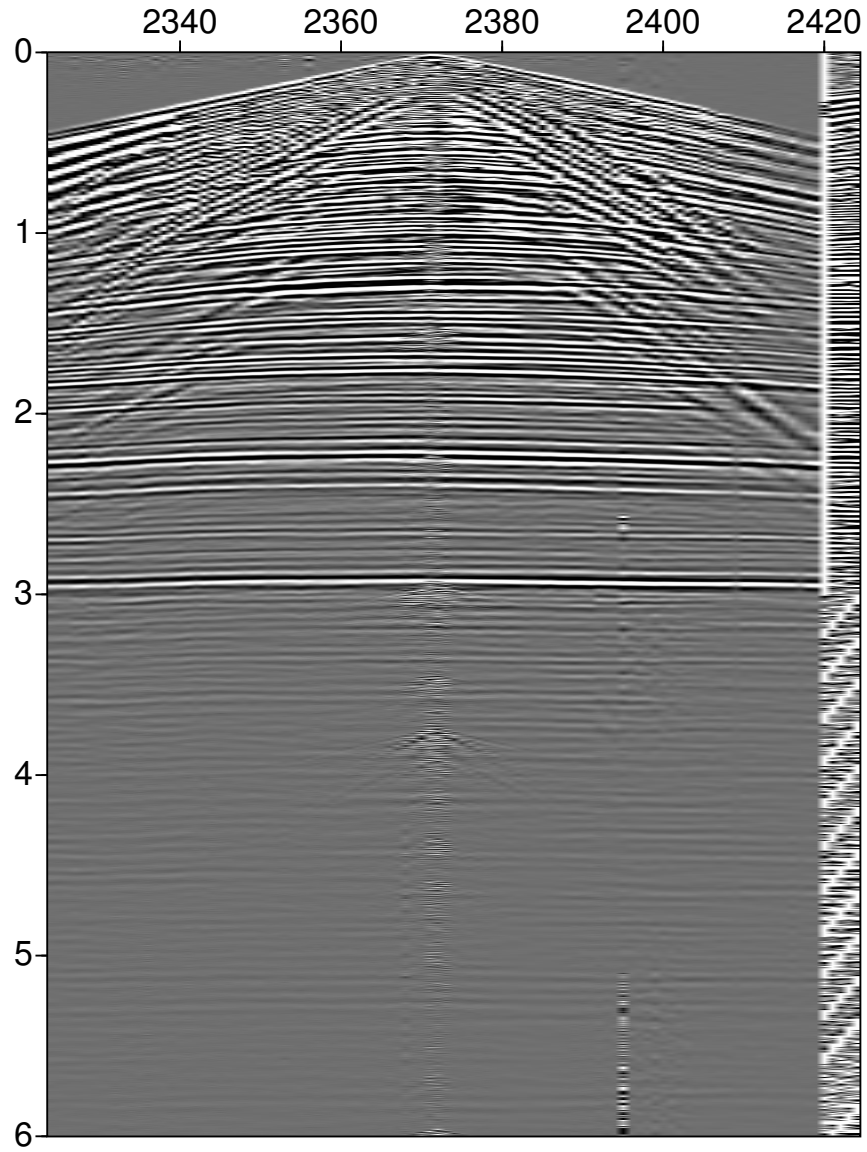


Figure 5: The 24th shot record without processing. The last 5 traces are auxiliary traces. There is noise on the traces closest to the shotpoint. One trace has a noise burst near the maximum time.

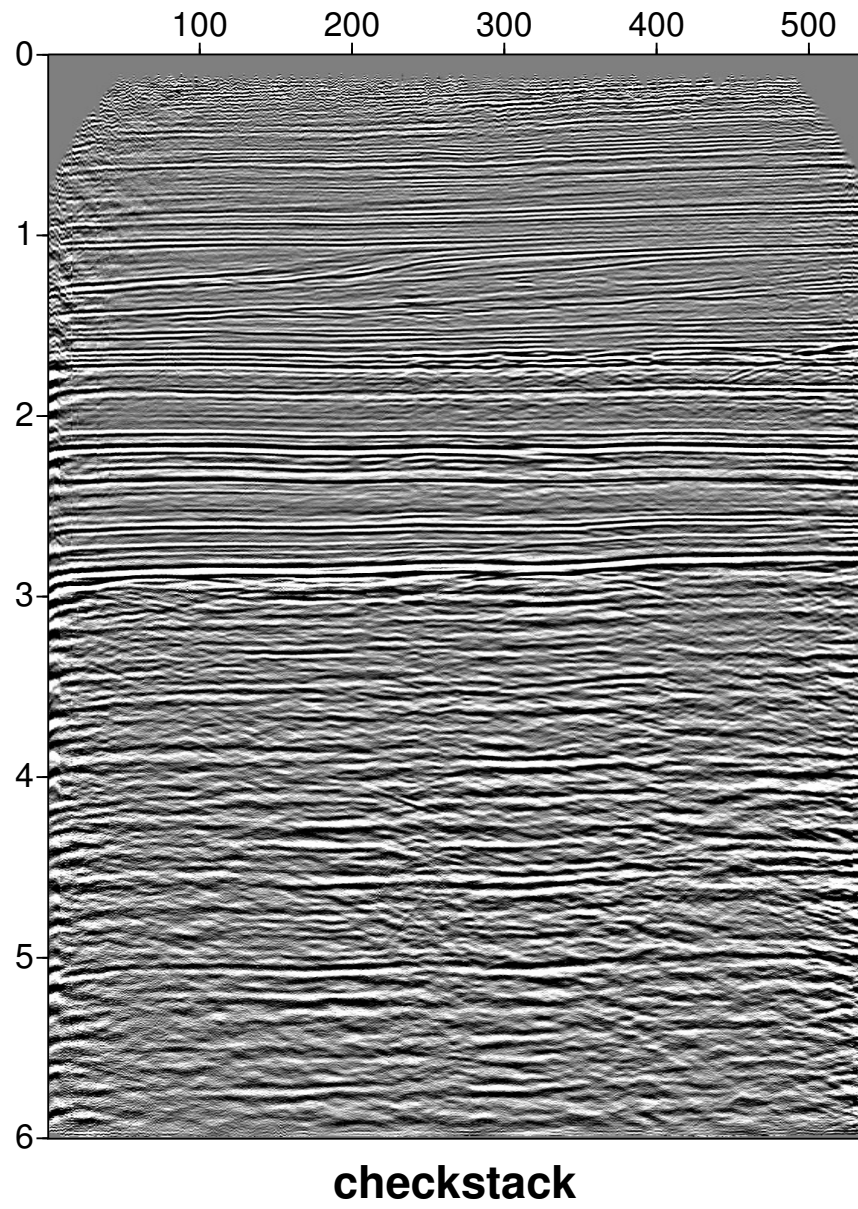


Figure 6: Plot of the SEG-Y of the final stack from the 1981 processing. **Schleicher** –

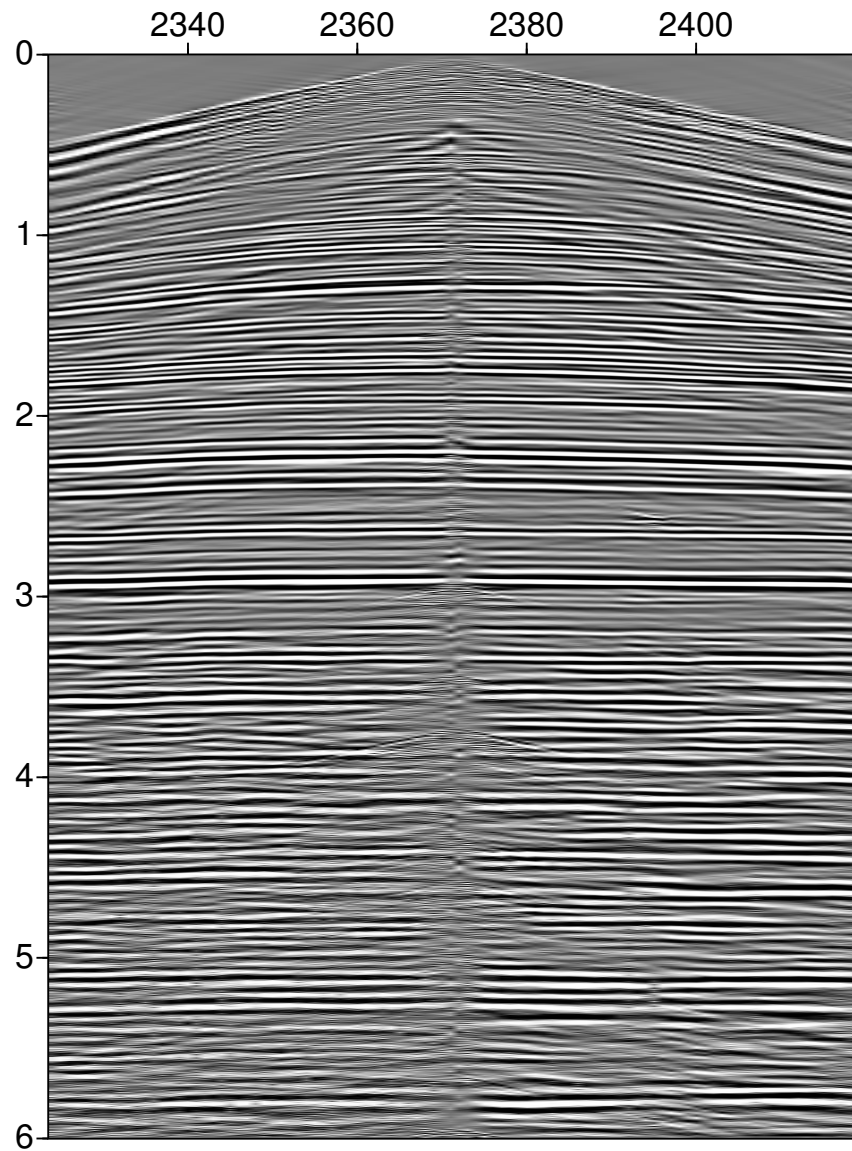


Figure 7: The 24th shot record after agc and velocity filter. **Schleicher** –

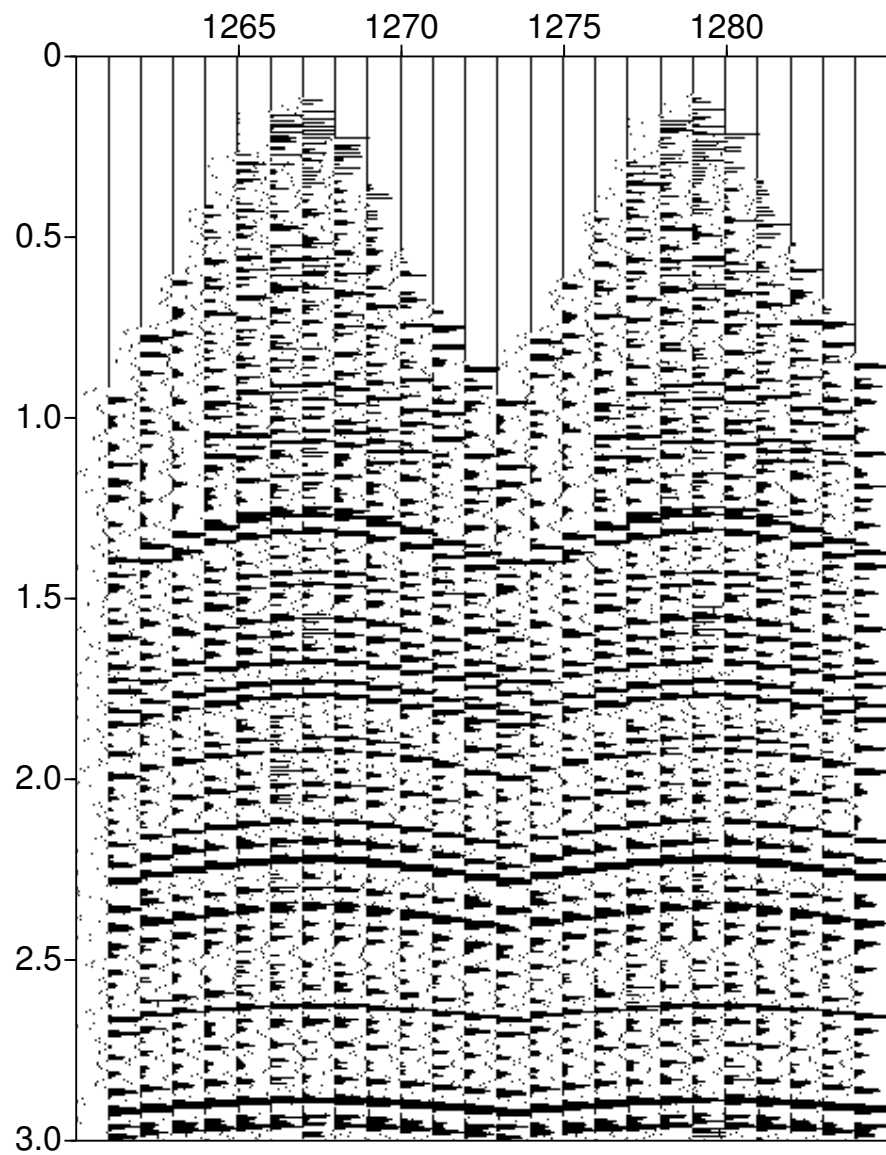


Figure 8: CDP gathers 250 and 251 with mute applied. **Schleicher** –

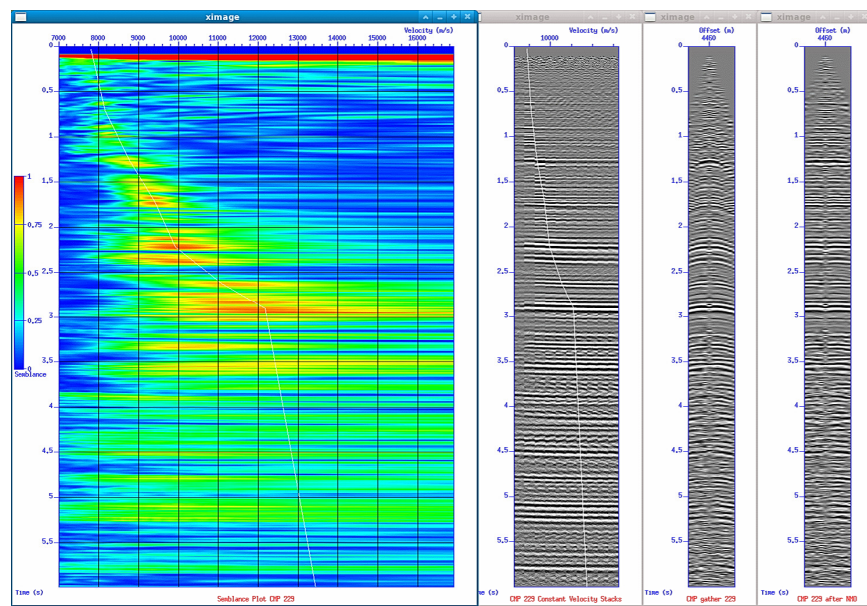


Figure 9: Example velocity analysis from iva.sh. **Schleicher** –

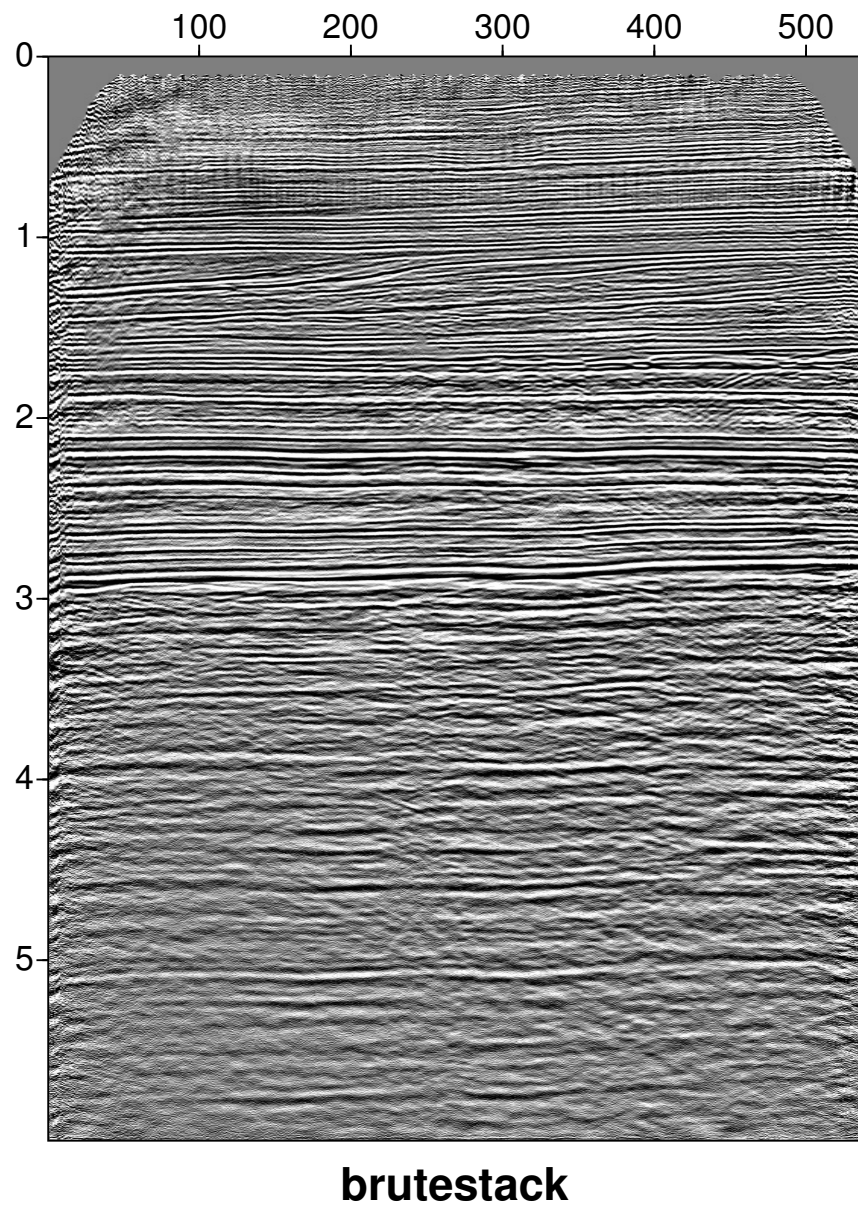


Figure 10: Brute stack.**Schleicher** –

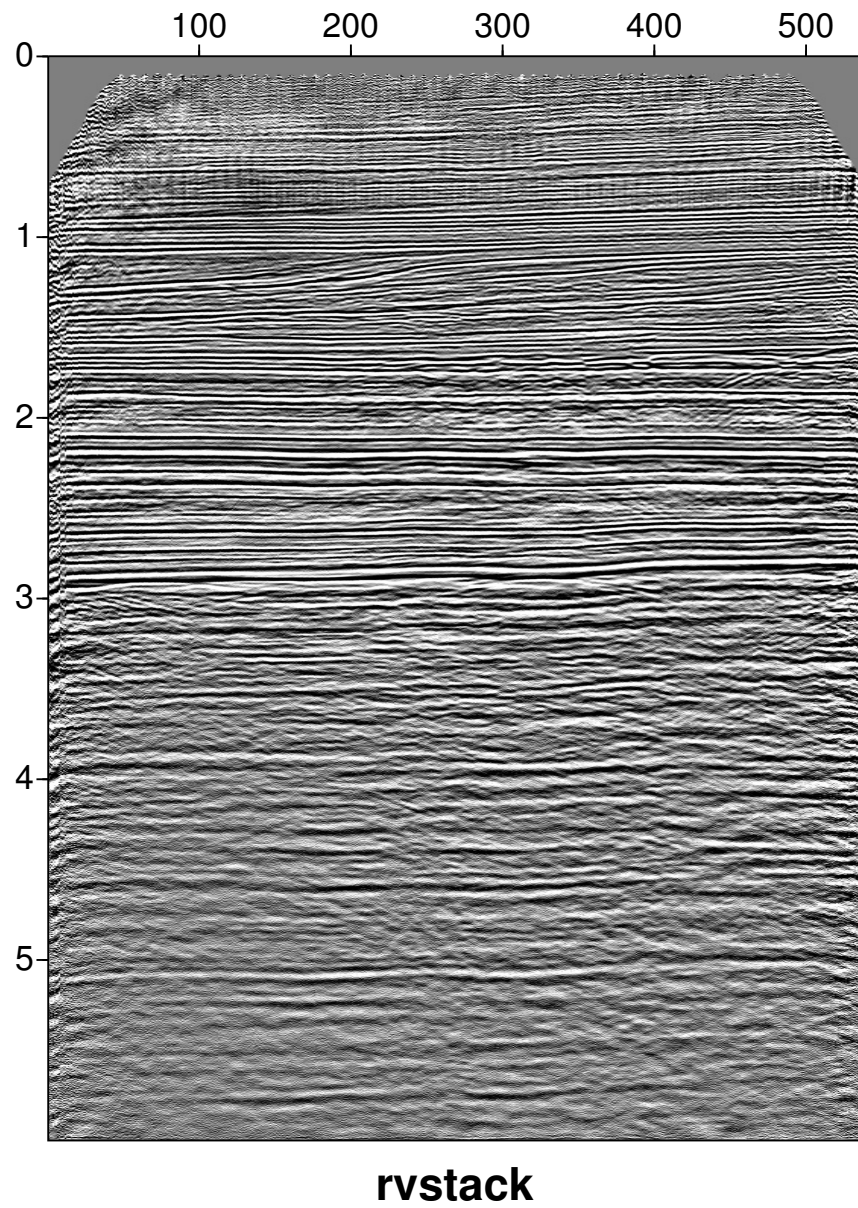


Figure 11: Stack with first pass residual statics and a lateral variable stacking velocity field.

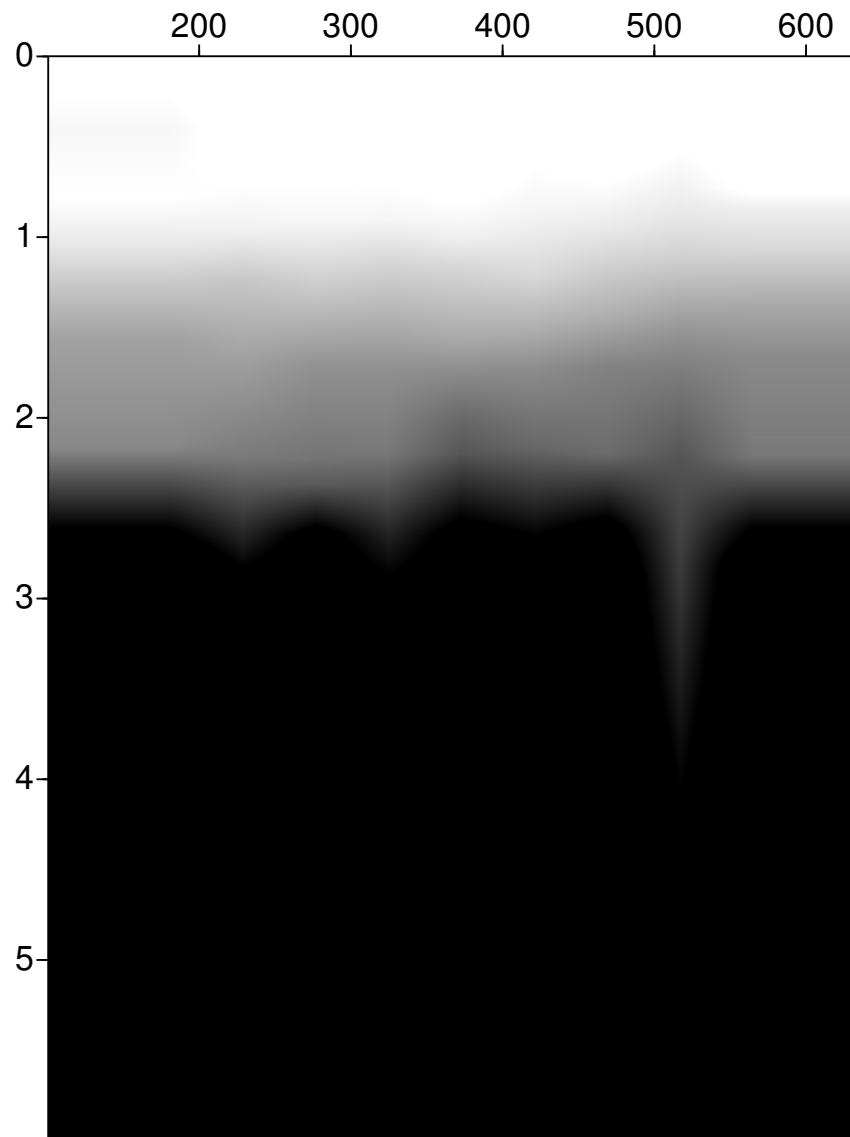


Figure 12: The lateral variable stacking velocity field interpreted after residual statics.



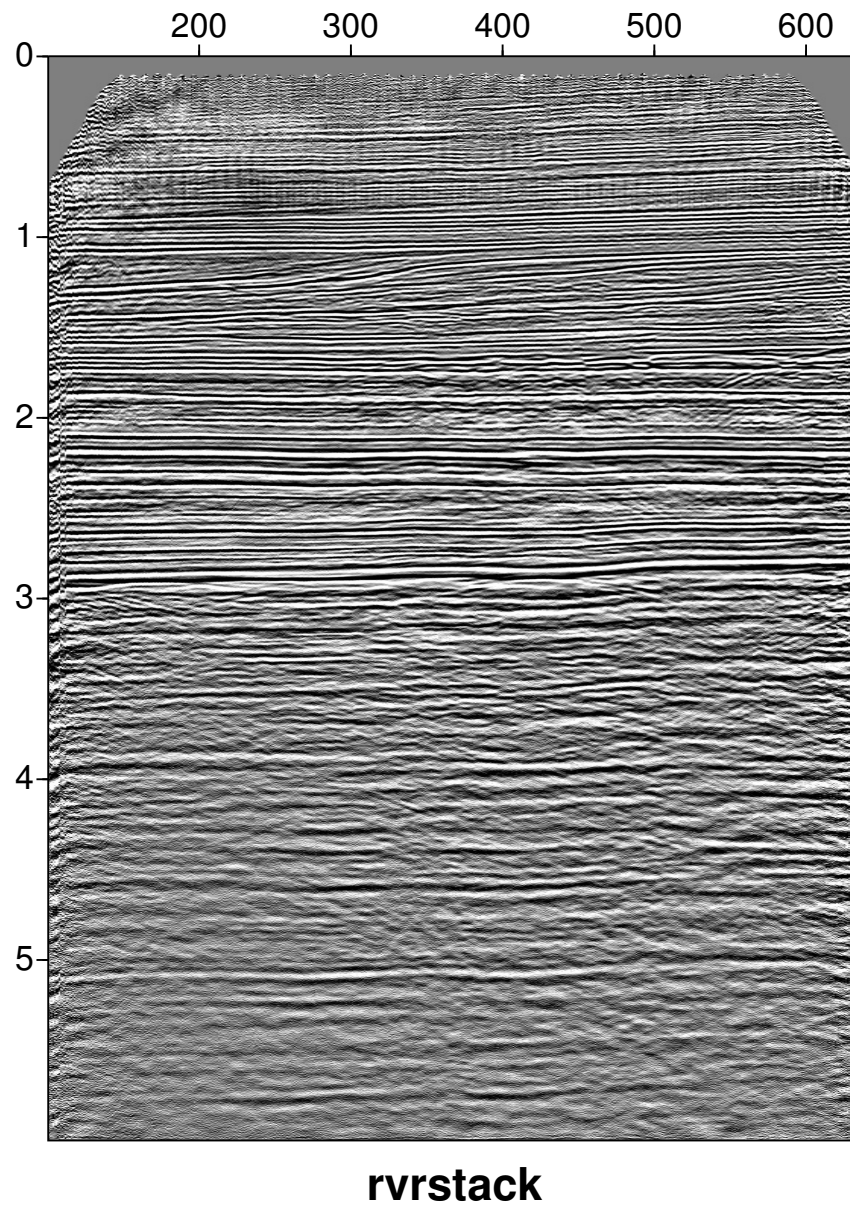


Figure 13: The final stack with the lateral variable stacking velocity and a second pass of residual statics.

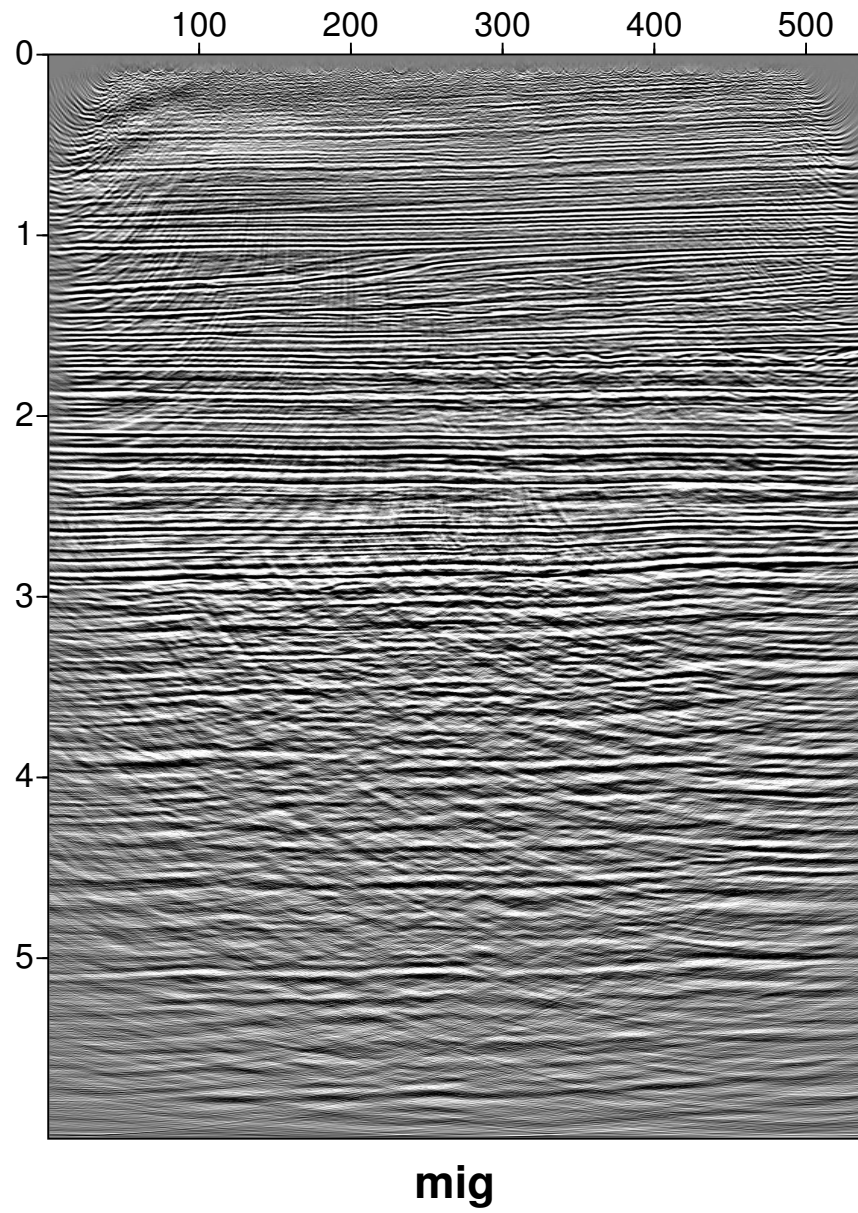


Figure 14: Post stack Kirchhoff migration of the final stack. **Schleicher** –