# GIS in Archaeology

## 10 - Least Cost Path Analysis

Martin Hinz

Institut für Archäologische Wissenschaften, Universität Bern

04/12/24

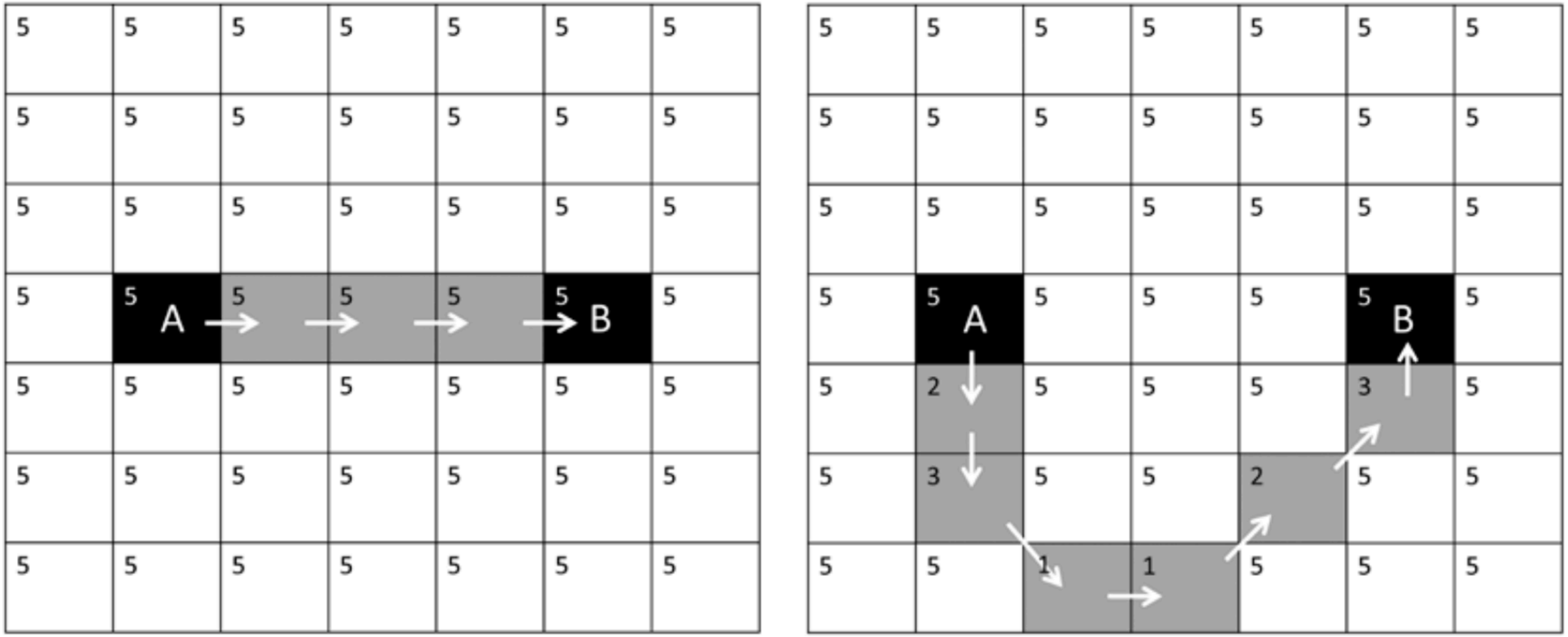
You can download a pdf of this presentation.

# What is it good for?

- Reconstruction of pathways
- Estimation of cost distances
- Estimation of interaction potential
- Archaeoprognosis
- Identification of factors that determine the course of known pathways
- Relative dating of sites
- Identification of central places
- Heritage Management: Modern road planning to minimise the impact on the perception of archaeological monuments.

# What is it good for?

The basic principle of least-cost-path analysis to determine optimal routes or interaction corridors (according to Surface-Evans/White 2012, 3 Fig. 1)



| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 A | 5 | 5 | 5 | 5 B | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 A | 5 | 5 | 5 | 5 B | 5 |
| 5 | 2 | 5 | 5 | 5 | 3 | 5 |
| 5 | 3 | 5 | 5 | 2 | 5 | 5 |
| 5 | 5 | 1 | 1 | 5 | 5 | 5 |

On the left: Movement through a geographical space from the starting point (A) to the destination (Z), which is determined by cells with equal costs (cost = 5). Right: Movement through a geographical space from the point of origin (A) to the destination (Z) determined by cells with different costs.

# What we need

- Data = e.g. digital terrain model, (find) points, vector data
- Costs = e.g. relief, vegetation, water, social and cultural aspects
- Cost functions = translation of benefits and obstacles to mobility into "locomotion effort" (energy consumption, speed, etc.)
- LCP algorithms = e.g. pathfinding algorithm through the cost grid

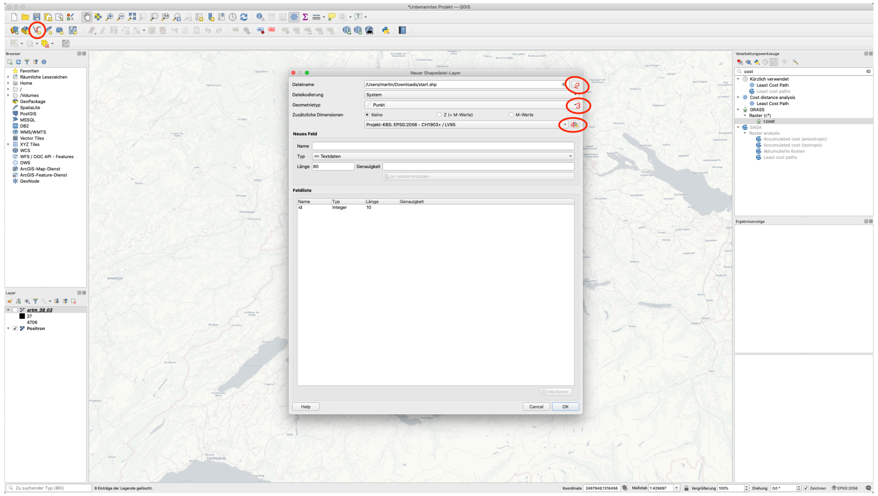
# Data

- a DEM
  - maybe a SRTM from https://dwtkns.com/srtm/
  - I suggest taking Switzerland
  - If you can not download from this source, you might try this link to ESA for the tile of switzerland.
- A start point
  - I suggest taking Bern
- A End Point
  - I suggest taking Basel

# Data Preparation (1)

We need a start and endpoint to calculate the path in individual layers.

- Start QGIS
- Add a basemap for reference (eg. Positron)
- create a new Vector point layer and call it 'start'
  - select storage location, geometry type point and CRS 2056
- add a point at Bern
- create a new Vector point layer and call it 'end' equivivalen to the start layer
- add a point at Basel
- Zoom in so that you can just see Basel and Bern



Dateiname: /Users/martin/Downloads/start.shp

Dateikodierung: System

Geometrietyp: Punkt

Zusätzliche Dimensionen: ● Keine ○ Z (+ M-Werte) ○ M-Werte

Projekt-KBS: EPSG:2056 - CH1903+ / LV95

Neues Feld

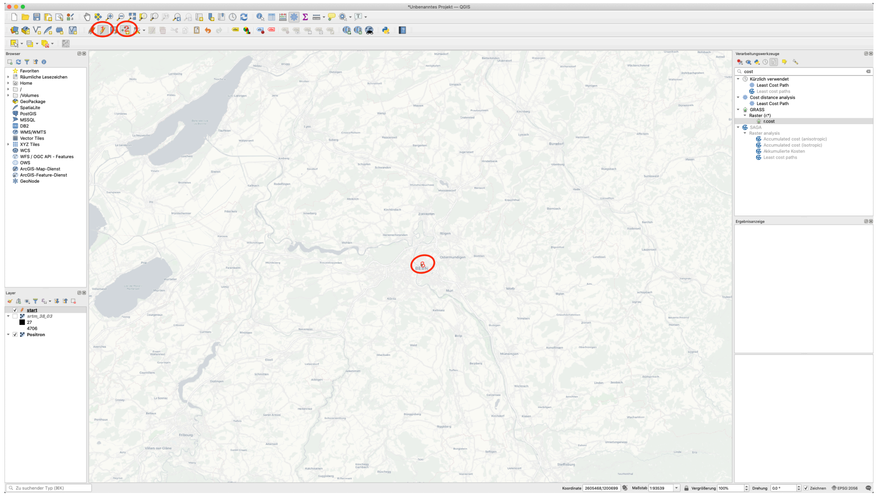
Name:

Typ: Textdaten

Länge: 80 Genauigkeit:

Feldliste

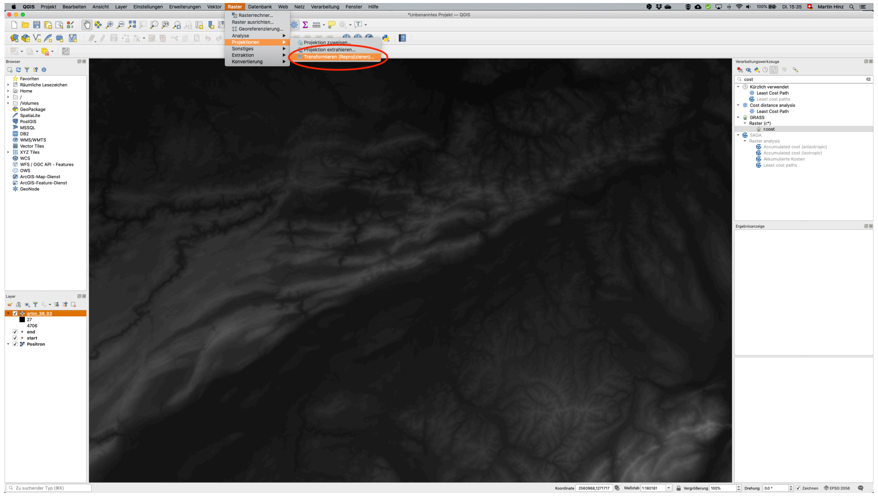
| Name | Typ | Länge | Genauigkeit |
|---|---|---|---|
| id | Integer | 10 | |

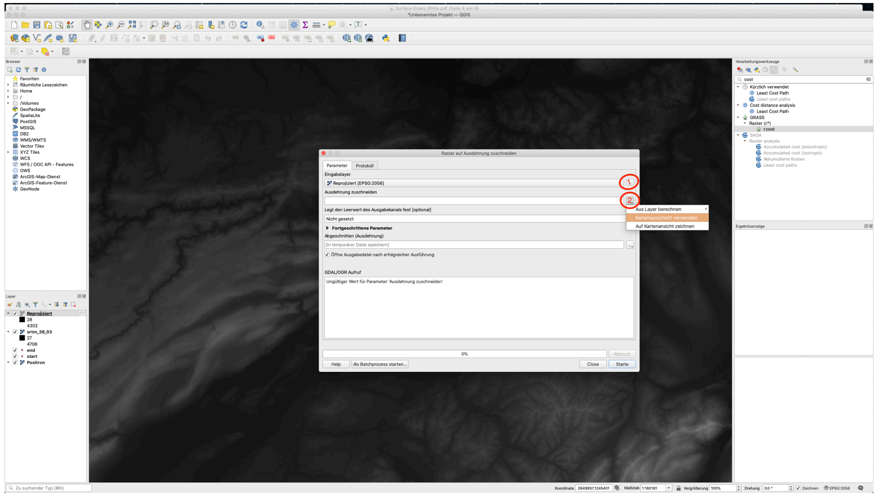


# Data Preparation (2)

We reduce the amount of data to be processed to speed things up.

- Download the tile
- Add the tile to QGIS
- Reproject the tile to epsg 2056
- Click on 'Raster > Extraction > Crop Raster to Extend'
- Select the Reprojected Raster
- Select Crop Extend from Map Extend





Raster auf Ausdehnung zuschneiden

Eingabelayer: Reprojiziert [EPSG:2056]

Ausdehnung zuschneiden

Legt den Leerwert des Ausgabekanals fest [optional]: Nicht gesetzt

Abgeschnitten (Ausdehnung): [In temporärer Datei speichern]

✓ Öffne Ausgabedatei nach erfolgreicher Ausführung

GDAL/OGR Aufruf

Ungültiger Wert für Parameter 'Ausdehnung zuschneiden'

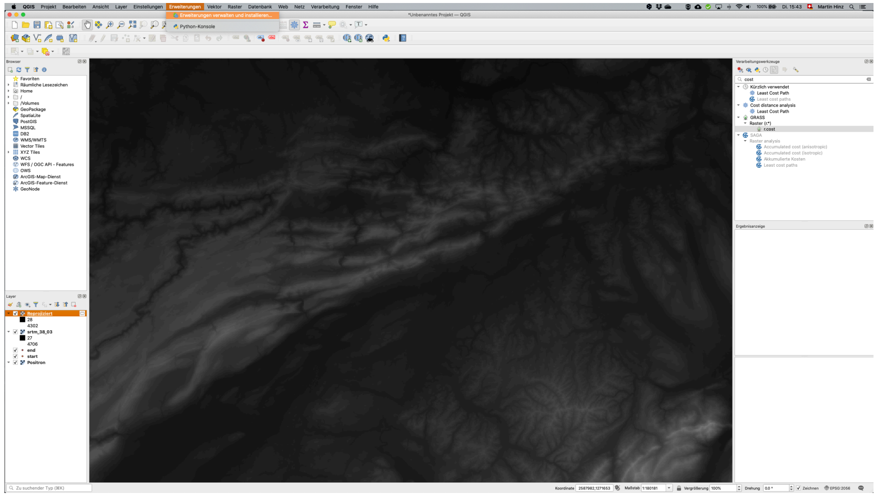
Parameter
Protokoll
1
2
Aus Layer berechnen
Kartenausschnitt verwenden
Auf Kartenansicht zeichnen
Fortgeschrittene Parameter
0%
Abbruch
Help
Als Batchprozess starten...
Close
Starte

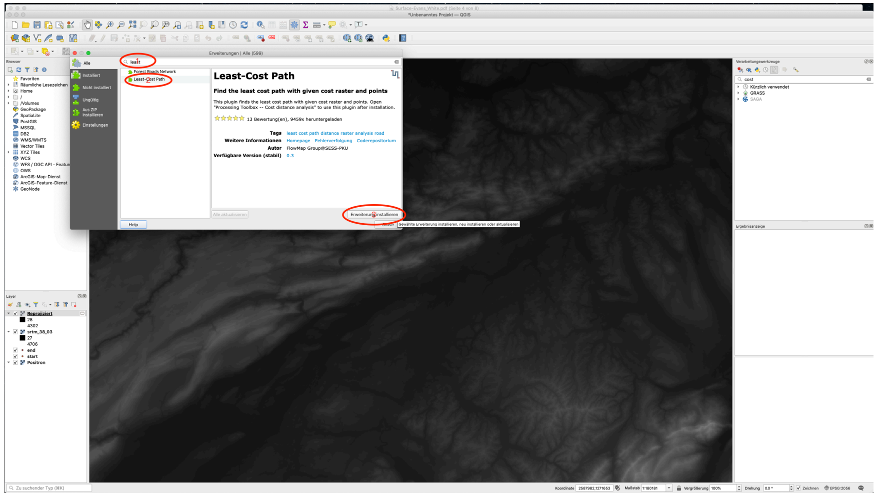
You now can turn of all layers except your cropped DEM and the start and endpoint.

# Quick and Dirty Least Cost Path

For a simple Least Cost Path, we use a plugin

- Open the Extension Manager
- Search for the Plugin 'Least Cost Path'
- Install





**Least-Cost Path**

**Find the least cost path with given cost raster and points**

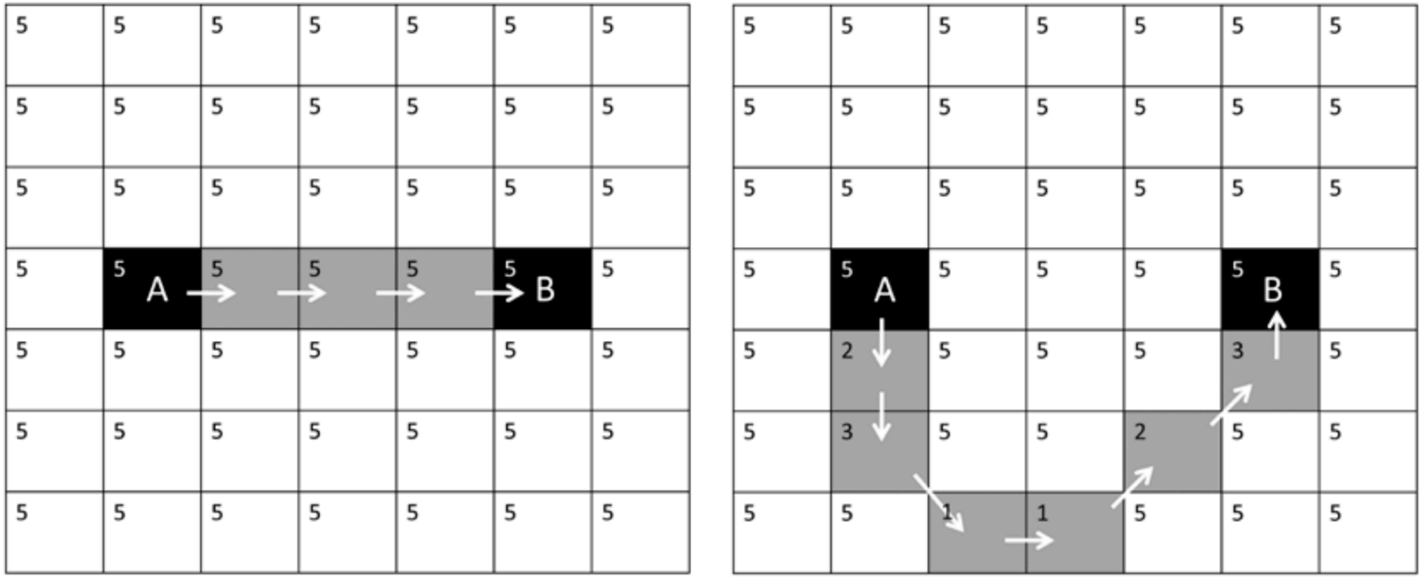
This plugin finds the least cost path with given cost raster and points. Open "Processing Toolbox -- Cost distance analysis" to use this plugin after installation.

13 Bewertung(en), 9459x heruntergeladen

**Tags** least cost path distance raster analysis road
**Weitere Informationen** Homepage Fehlerverfolgung Coderepositorium
**Autor** FlowMap Group@SESS-PKU
**Verfügbare Version (stabil)** 0.3

# Costs and Cost Grid

We are looking for the fastest path. Remember:



| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 A | 5 | 5 | 5 | 5 B | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 A | 5 | 5 | 5 | 5 B | 5 |
| 5 | 2 | 5 | 5 | 5 | 3 | 5 |
| 5 | 3 | 5 | 5 | 2 | 5 | 5 |
| 5 | 5 | 1 | 1 | 5 | 5 | 5 |

- On a plain, the fastest path is a straight line between to points
- In a terrain, the fastest path depends on the terrain
- If we ignore surface quality, it is dependend on slope

# Cost functions in respect to Slope

## 'Ox cart function' according to Herzog 2012

$$f(s) = 1 + (s/c)^2$$

s = Slope in percent; c = critical slope = 10 bis 12 %

## Energy consumption function for pedestrians according to Herzog 2012

$$f(s) = 1337,8s^6 + 278,19s^5 - 517,39s^4 - 78,199s^3 + 93,419s^2 + 19,825s + 1,64$$

s = Slope in percentage / 100

## Inverse speed function for pedestrians according to Tobler 1993

$$f(s; Tobleraway) = 1/(6^{-3,5*abs(S+0,05)})$$

$$f(s; Toblertowards) = 1/(6^{-3,5*abs(S-0,05)})$$

$$f(s; ToblerMittelwert) = (f(s; Tobleraway) + f(s; Toblertowards))/2$$

# Cost functions in respect to Slope

## Why does in matter?

- Different modes of transport are differently susceptible to slope
- Different Functions produce different costs dependend on the slope
- In the large scale, this can influence your results dramaticaly
- There is a whole body of literature on this topic...

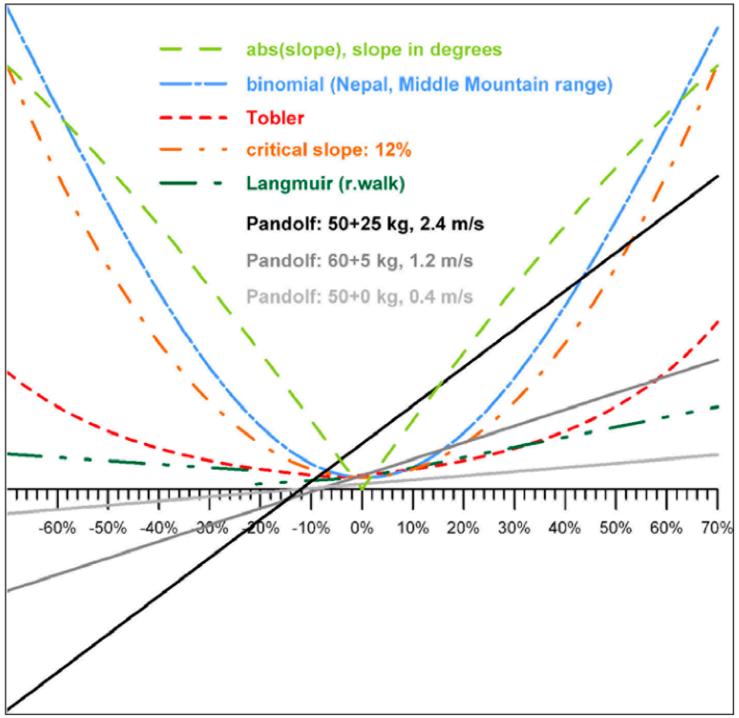


Fig. 3 – Comparison of slope-dependent cost functions used in the case studies. The cost functions are scaled to assist comparisons. The Pandolf cost functions are negative for steep downhill slopes.

*Herzog 2014*

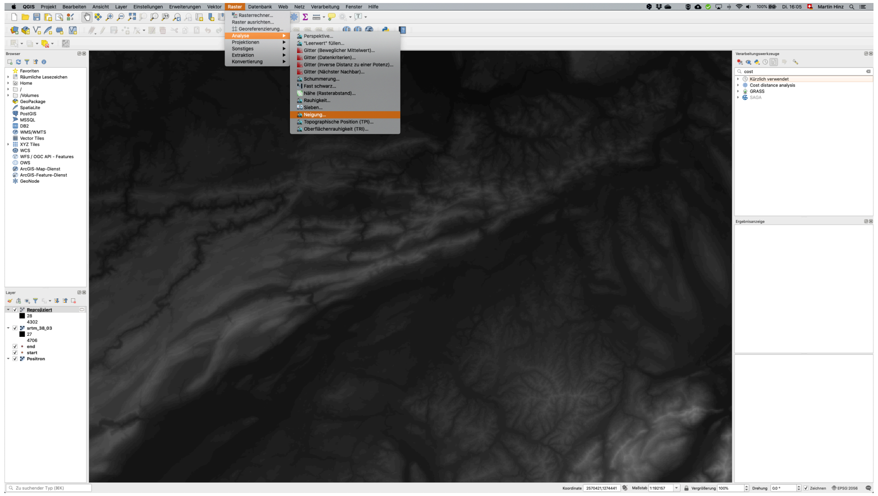
# Cost functions in respect to Slope

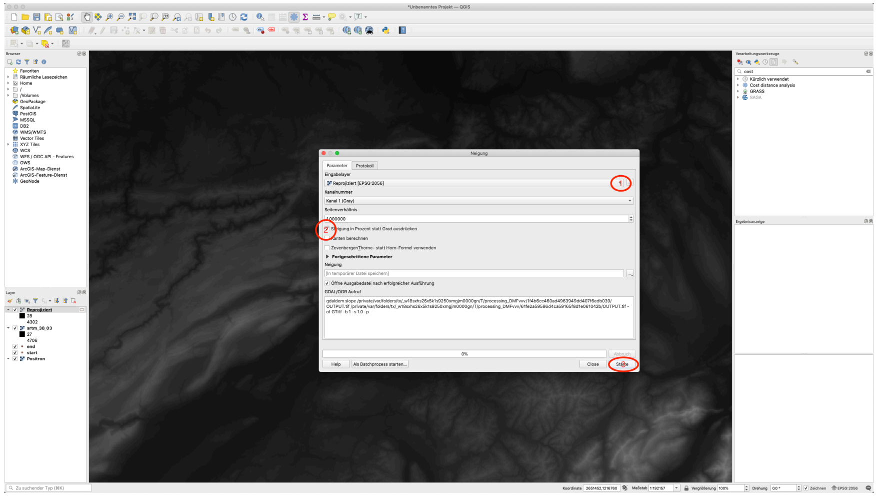
## Intimitated?

- We use the simple ox cart function
- Consider using a more complicated function

## First: Calculate Slope

- You still now how from last week?
- **We will need percentage!**





Eingabelayer: Reprojiziert [EPSG:2056]

Kanalnummer: Kanal 1 (Gray)

Seitenverhältnis: 1.000000

☑ Steigung in Prozent statt Grad ausdrücken

□ Kanten berechnen

□ ZevenbergenThorne- statt Horn-Formel verwenden

Fortgeschrittene Parameter

Neigung: [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

GDAL/OGR Aufruf

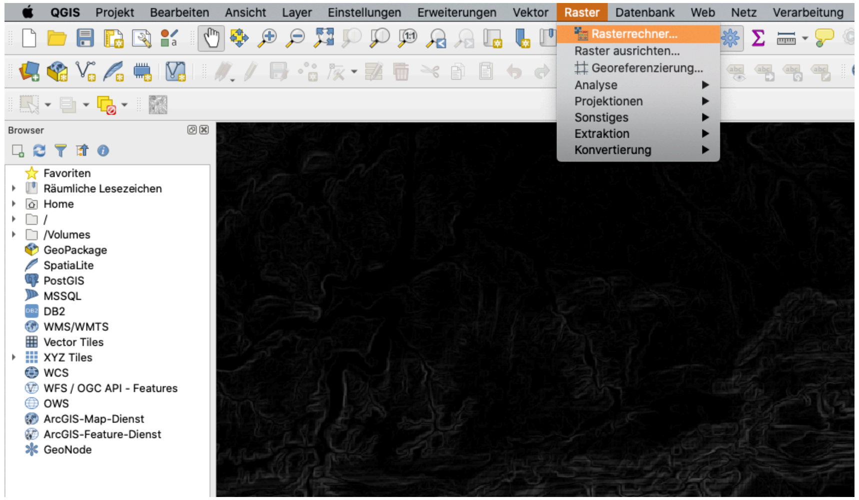
```
gdaldem slope /private/var/folders/tx/_w18sxhs26x5k1s9250xmgjm0000gn/T/processing_DMFvvv/1f4b6cc460ad4963949dd407f6edb039/
OUTPUT.tif /private/var/folders/tx/_w18sxhs26x5k1s9250xmgjm0000gn/T/processing_DMFvvv/61fe2a59586d4ca59165f8d1e061042b/OUTPUT.tif -
of GTiff -b 1 -s 1.0 -p
```

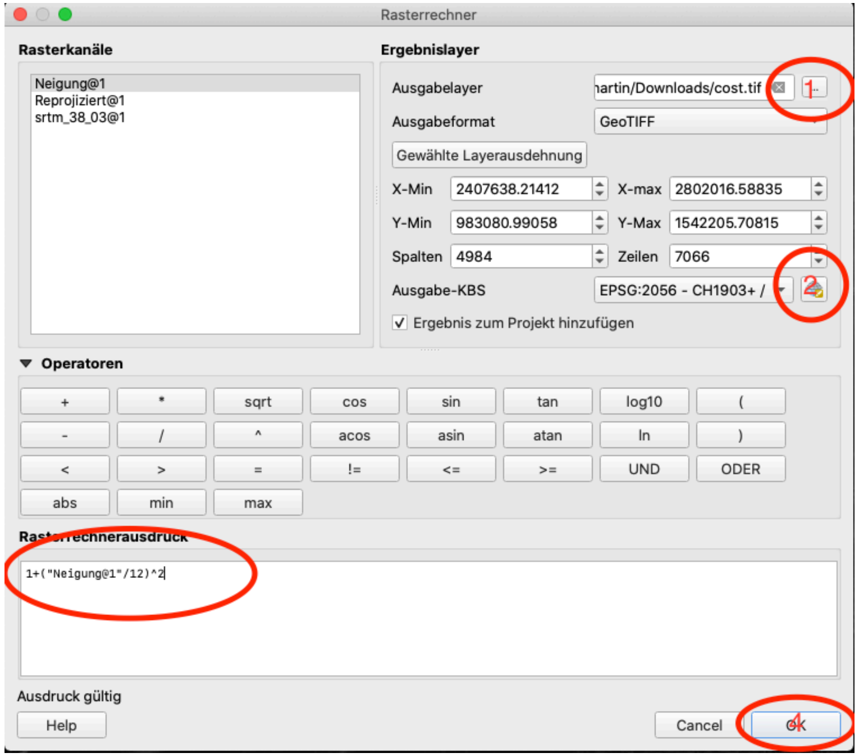
# Calculating Costs from Slope

- We use the Raster Calculator of QGIS
- We just plugin the ox cart function like below:

$$f(s) = 1 + (s/c)^2$$

1+(Slope/12)^2





Rasterkanäle:
Neigung@1
Reprojiziert@1
srtm_38_03@1

Ausgabelayer: artin/Downloads/cost.tif

Ausgabeformat: GeoTIFF

X-Min 2407638.21412 X-max 2802016.58835

Y-Min 983080.99058 Y-Max 1542205.70815

Spalten 4984 Zeilen 7066

Ausgabe-KBS EPSG:2056 - CH1903+ /

☑ Ergebnis zum Projekt hinzufügen

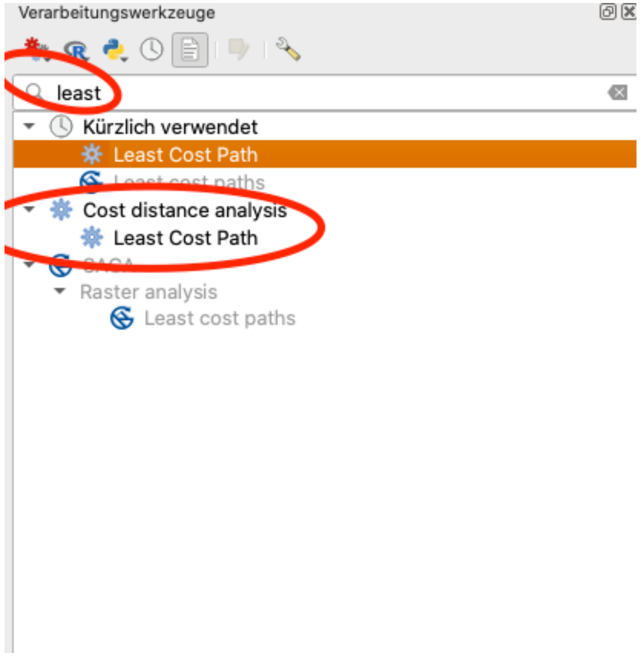
Rasterrechnerausdruck

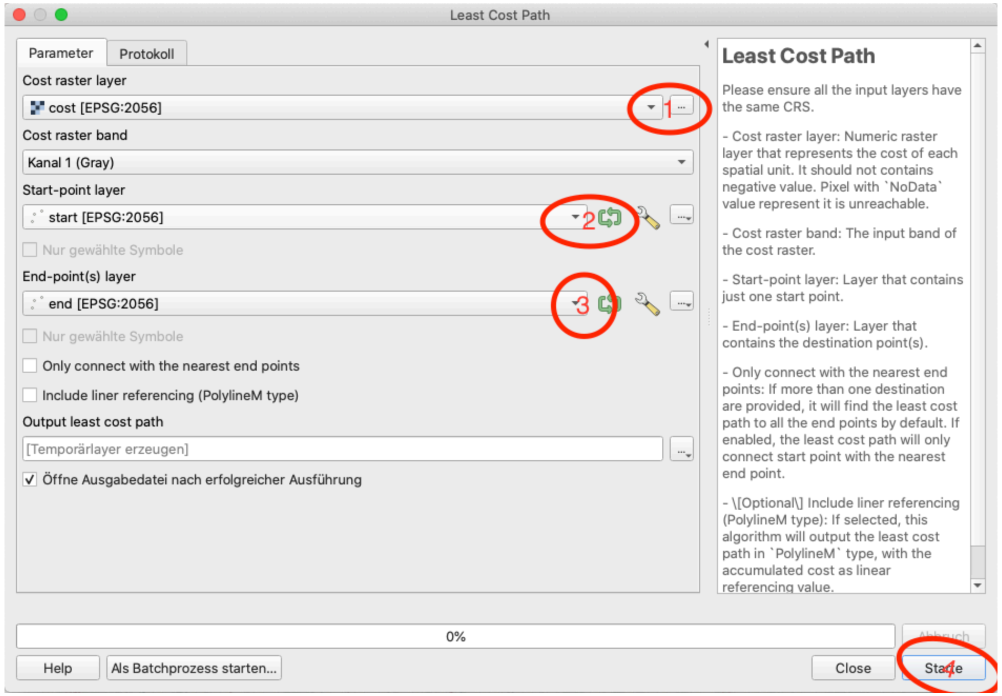
```
1+("Neigung@1"/12)^2
```

Ausdruck gültig

# Lets calculate!

- Open the Toolbox ⚙
- Search for 'least cost'
- Select the 'Least Cost Path' tool from 'Cost Distance Analysis'
  - **Do not choose the SAGA tool, it works differently**
- Select input cost layer, start and end point layer
- Click on 'Run' (this takes a while)





Cost raster layer: cost [EPSG:2056]

Cost raster band: Kanal 1 (Gray)

Start-point layer: start [EPSG:2056]

□ Nur gewählte Symbole

End-point(s) layer: end [EPSG:2056]

□ Nur gewählte Symbole

□ Only connect with the nearest end points

□ Include liner referencing (PolylineM type)

Output least cost path: [Temporärlayer erzeugen]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

**Least Cost Path**

Please ensure all the input layers have the same CRS.

- Cost raster layer: Numeric raster layer that represents the cost of each spatial unit. It should not contains negative value. Pixel with `NoData` value represent it is unreachable.

- Cost raster band: The input band of the cost raster.

- Start-point layer: Layer that contains just one start point.

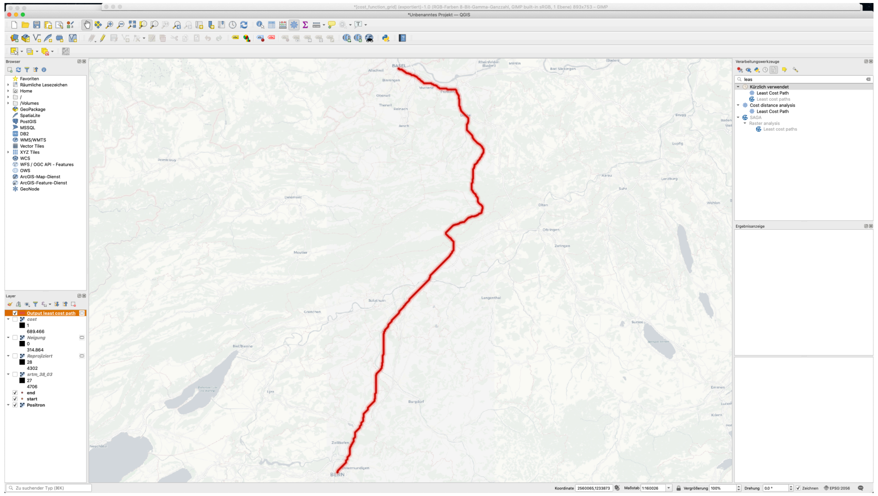
- End-point(s) layer: Layer that contains the destination point(s).

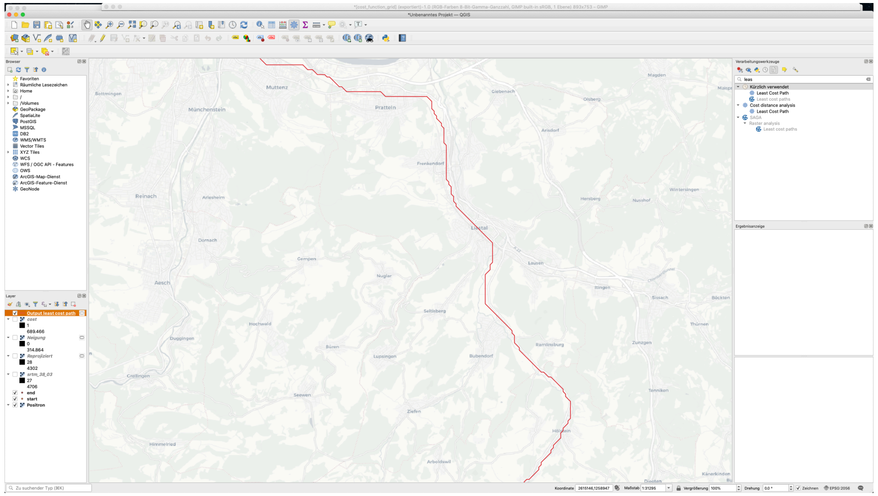
- Only connect with the nearest end points: If more than one destination are provided, it will find the least cost path to all the end points by default. If enabled, the least cost path will only connect start point with the nearest end point.

- \[Optional\] Include liner referencing (PolylineM type): If selected, this algorithm will output the least cost path in `PolylineM` type, with the accumulated cost as linear referencing value.

# Result

- Turn all layers of except the points, the least cost path line and the basemap
- Compare the resulting Least Cost Path with the road network
    - On the plain, the LCP is more straight than the road
    - In the Hills, Road and LCP are more or less identical
- The more defined the terrain is, the more the least cost path is determined by it

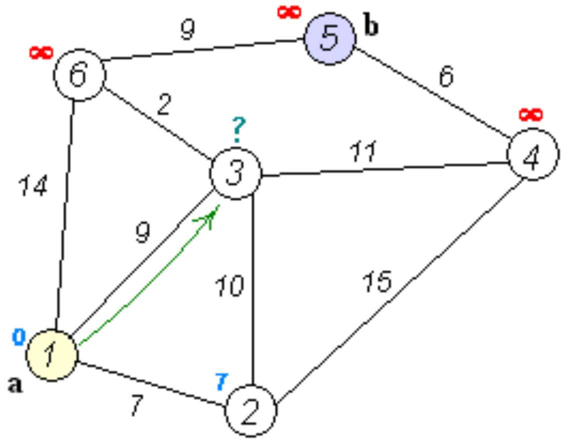




# Dijkstra-Algorithm

> The basic idea of the algorithm is to always follow the edge that promises the shortest route section from the start node. Other edges are only followed if all shorter route sections (also beyond other nodes) have been considered. This procedure ensures that when a node is reached, no shorter path to it can exist. - wikipedia

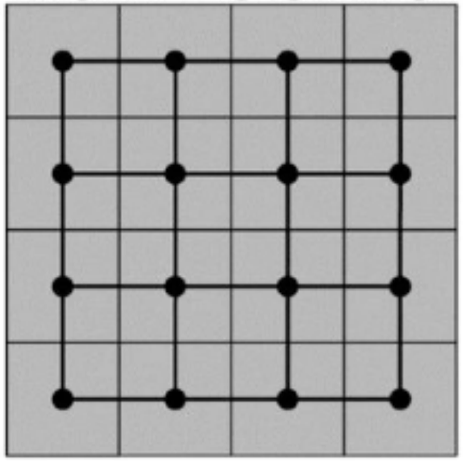
1. find the most cost-effective step from the starting point
2. note the cost and mark the destination of the step.
3. find the cheapest step from a visited cell adjacent to an unused cell
4. note the costs and the starting point of the step and the destination
5. repeat 3 and 4 until the target point is reached.
6. reconstruct the best route by stringing together the best connections from the destination to the starting point.(after Oliver Nakoinz)



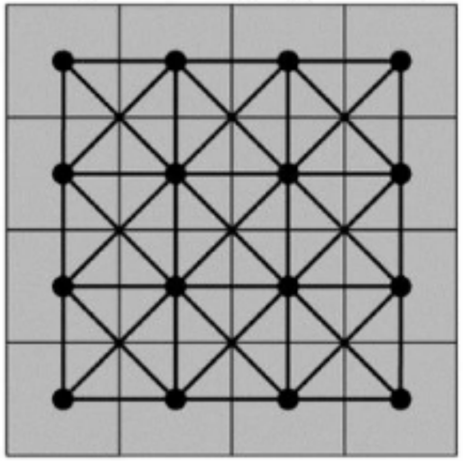
# Some Movement Directions

More possible directions -> more precise results, but also more computational time

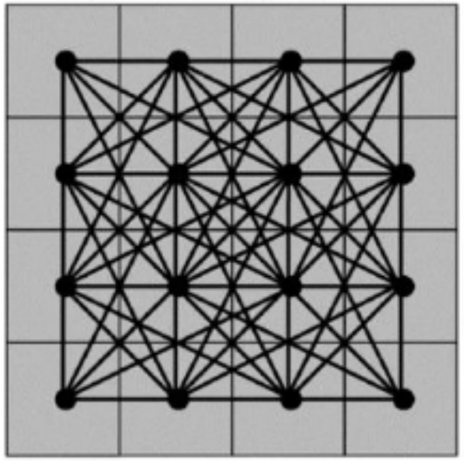
Neumann Nachbarschaft
4 mögliche Bewegungsrichtungen



King´s Move
8 mögliche Bewegungsrichtungen



King´s und Knight´s Moves
16 mögliche Bewegungsrichtungen

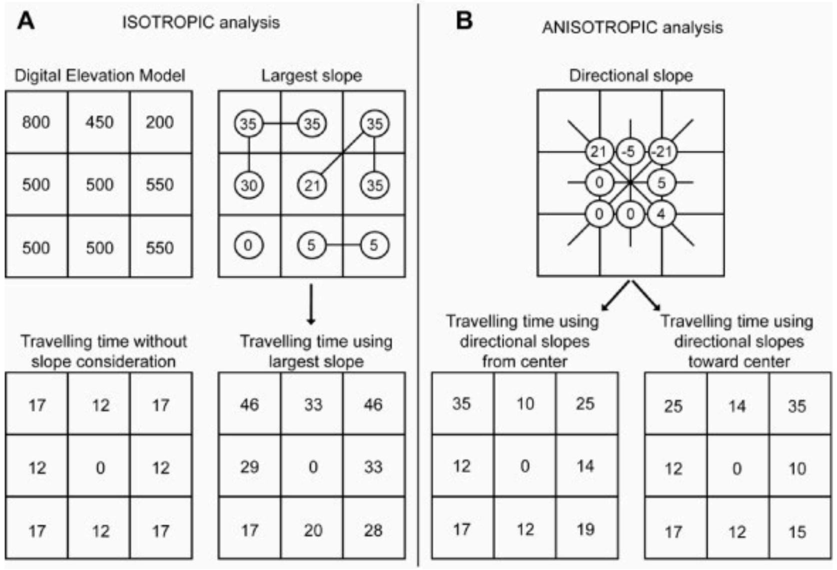


# Isotropic vs. anisotropic Analysis

- Costs are calculated per cell
- if you move along a slope, you actually walk on an even elevation
- if the movement direction is not considered, it is isotropic
- if they are taken into account, it is anisotropic
    - more accurate, but more calculation intensive
    - travel direction matters: The path from a -> b can be different than the path from b->a

The cost raster is isotropic, the plugin uses Dijkstra and Manhattan (Neumann) Neighborhood. There is a way independent from the plugin using SAGA, but we will not cover this here...

Maintain Elevation
Gain Elevation



A ISOTROPIC analysis

Digital Elevation Model

| | | |
|---|---|---|
| 800 | 450 | 200 |
| 500 | 500 | 550 |
| 500 | 500 | 550 |

Largest slope

Travelling time without slope consideration

| | | |
|---|---|---|
| 17 | 12 | 17 |
| 12 | 0 | 12 |
| 17 | 12 | 17 |

Travelling time using largest slope

| | | |
|---|---|---|
| 46 | 33 | 46 |
| 29 | 0 | 33 |
| 17 | 20 | 28 |

B ANISOTROPIC analysis

Directional slope

Travelling time using directional slopes from center

| | | |
|---|---|---|
| 35 | 10 | 25 |
| 12 | 0 | 14 |
| 17 | 12 | 19 |

Travelling time using directional slopes toward center

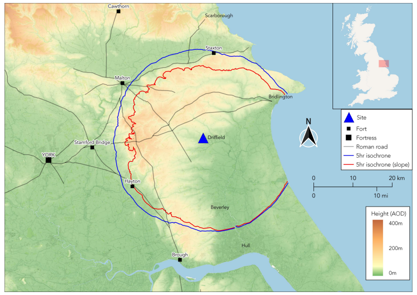
| | | |
|---|---|---|
| 25 | 14 | 35 |
| 12 | 0 | 10 |
| 17 | 12 | 15 |

*Source: David Lewis; Ray/Ebener 2009*

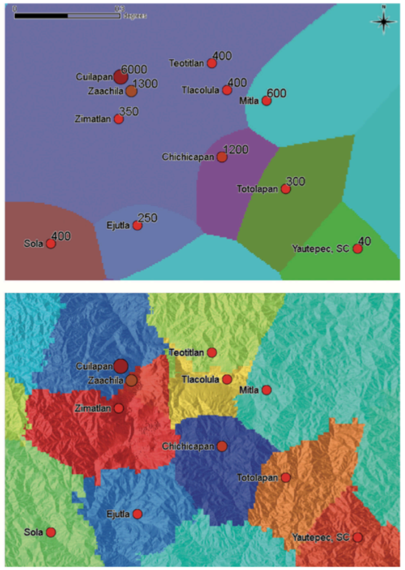
# Calculating walking time from a given start point

If you want to know, which points can be reached in what or a given time

- calculates the walking time in any direction
- if a maximum time is given, it is possible to determine the area reachable within this time
- can be used to estimate a territory used by a settlement
- can be based on terrain and other cost changing aspects (roads, barriers)



*Comparison between 5 hour isochrones, with and without slope. Source: http://www.chrismapsthepast.com*



*Comparison between a Thiessen-Polygon and an Cost-Defined (XTENT) model of the territories of the Maya lowland. Source: Ducke/Kroefges 2007*

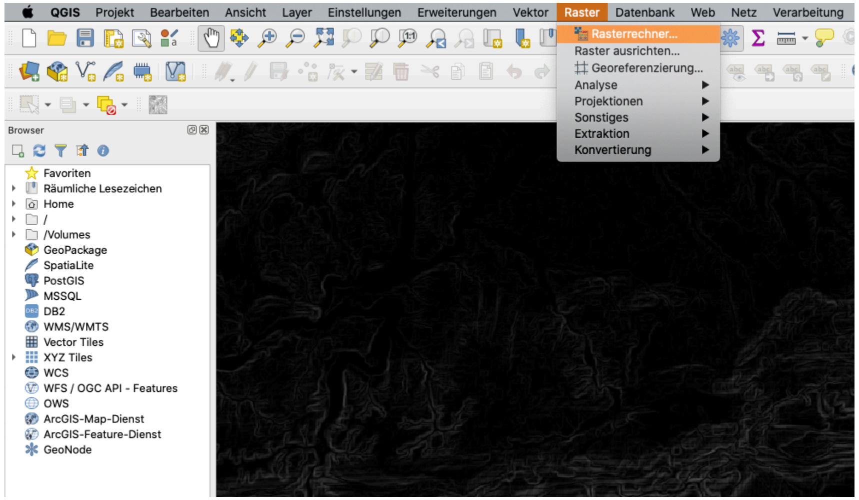
# Calculating in QGIS - Prerequisists

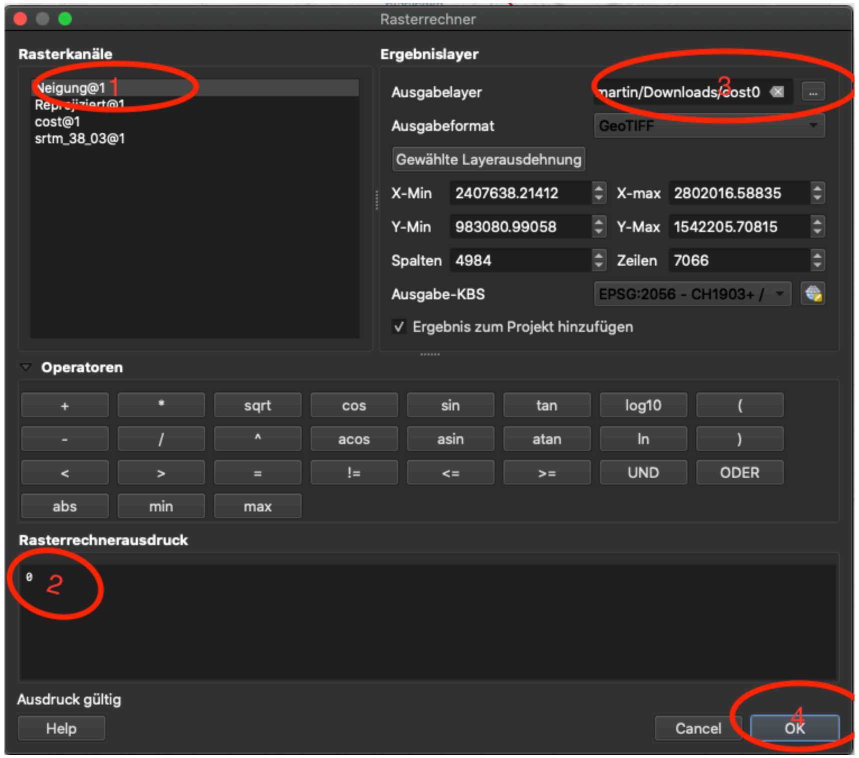
We need to utilize another GIS inside QGIS: GRASS.

- There is a function r.walk that calculates walking time.
- You can parameterise it for different Walking Cost functions, we work with the default (although it might be not optimal)
- It expects a start point, an elevation model and a 'friction cost' layer
    - With the friction cost you can introduce other costs beside the slope
    - **This is not the cost layer we just calculated**
    - To make our walking time only dependend on the slope, have to define a neutral raster layer containing only zeros (0)
    - We can use the raster calculator for this

# Creating a empty (zero) layer

- Start the raster calculator
- Select the Slope layer as template
- Write '0' in the 'Expression' pane
- Specify the output file, name it eg. 'cost0'
- Click on OK





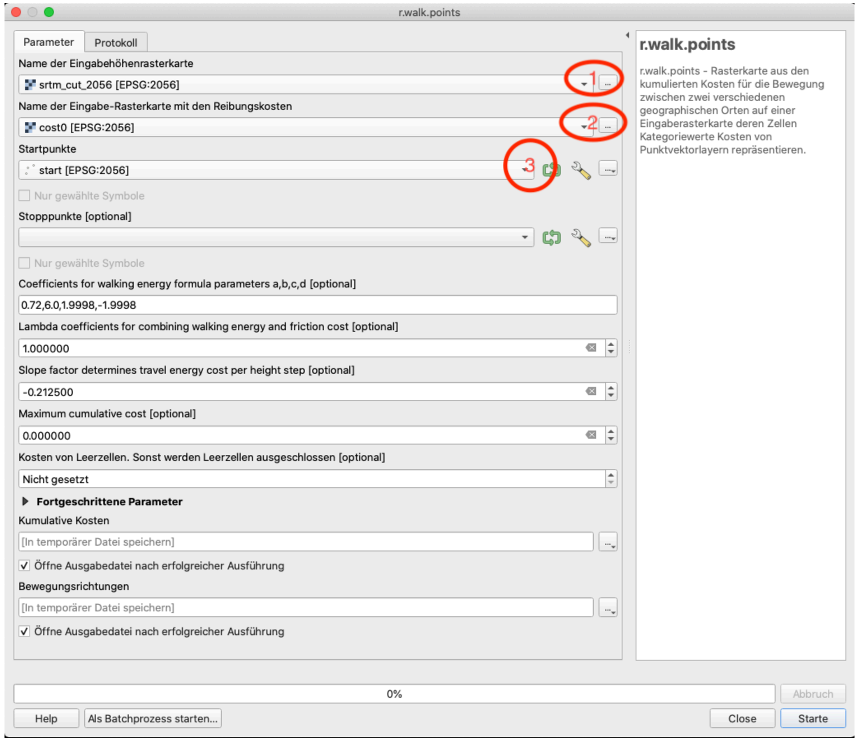
Rasterkanäle
Neigung@1
Reprojiziert@1
cost@1
srtm_38_03@1

Ergebnislayer
Ausgabelayer: martin/Downloads/cost0
Ausgabeformat: GeoTIFF
Gewählte Layerausdehnung
X-Min: 2407638.21412 X-max: 2802016.58835
Y-Min: 983080.99058 Y-Max: 1542205.70815
Spalten: 4984 Zeilen: 7066
Ausgabe-KBS: EPSG:2056 - CH1903+ /
✓ Ergebnis zum Projekt hinzufügen

Rasterrechnerausdruck
```
0
```
Ausdruck gültig

# Calculating in QGIS - Actual Calculation

- Open the Toolbox
- Search for 'r.walk' and open the 'r.walk.points' tool
- Select DEM, cost0 and start layer
- [you could define a stop point, where the maximum costs will be reached]
- [you also can define the formula for the walking function. It defaults to Langmuir]
- [in the advanced settings, you can specify using 'Kings move', it defaults to Manhattan move]
- Click on Run



Name der Eingabehöhenrasterkarte: srtm_cut_2056 [EPSG:2056]

Name der Eingabe-Rasterkarte mit den Reibungskosten: cost0 [EPSG:2056]

Startpunkte: start [EPSG:2056]

□ Nur gewählte Symbole

Stopppunkte [optional]

□ Nur gewählte Symbole

Coefficients for walking energy formula parameters a,b,c,d [optional]: 0.72,6.0,1.9998,-1.9998

Lambda coefficients for combining walking energy and friction cost [optional]: 1.000000

Slope factor determines travel energy cost per height step [optional]: -0.212500

Maximum cumulative cost [optional]: 0.000000

Kosten von Leerzellen. Sonst werden Leerzellen ausgeschlossen [optional]: Nicht gesetzt

▶ **Fortgeschrittene Parameter**

Kumulative Kosten: [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

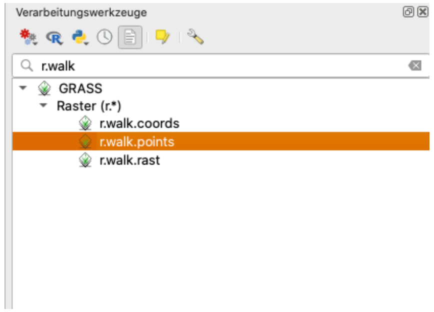
Bewegungsrichtungen: [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

0%

**r.walk.points**

r.walk.points - Rasterkarte aus den kumulierten Kosten für die Bewegung zwischen zwei verschiedenen geographischen Orten auf einer Eingaberasterkarte deren Zellen Kategoriewerte Kosten von Punktvektorlayern repräsentieren.



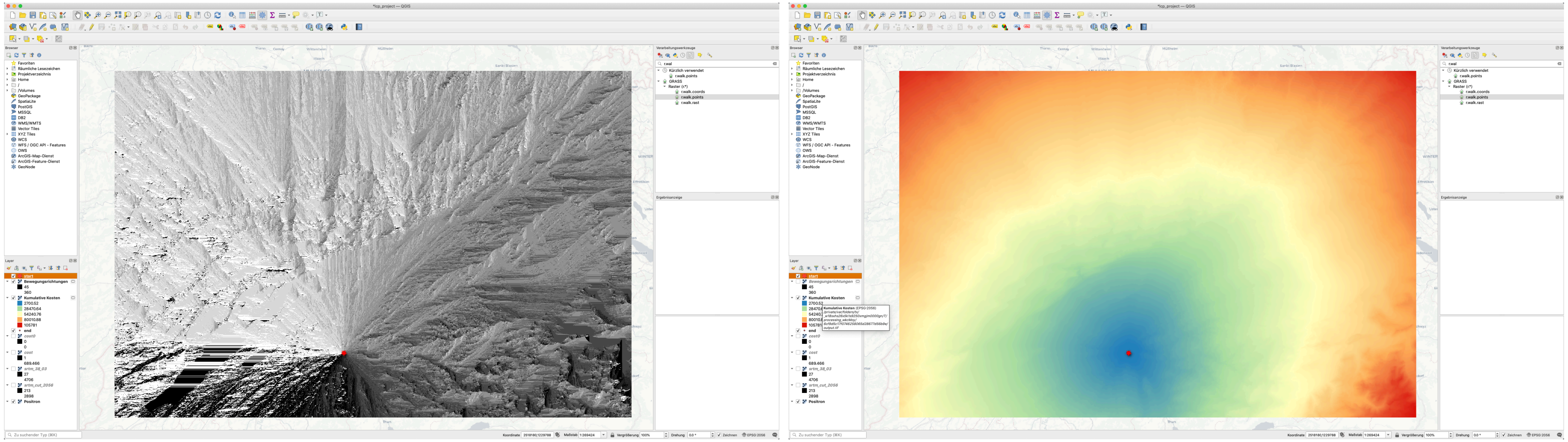
r.walk

- GRASS
  - Raster (r.*)
    - r.walk.coords
    - r.walk.points
    - r.walk.rast

# r.walk results

You get two resulting layers:

- Movement directions contains the movement choosen for the calculation at each raster cell
- More relevant is 'Cumulative Costs'
  - Here, the raster holds the walking time to the pixel cell from the start point measured in seconds
  - You can color this using pseudocolor, spectral, inverse



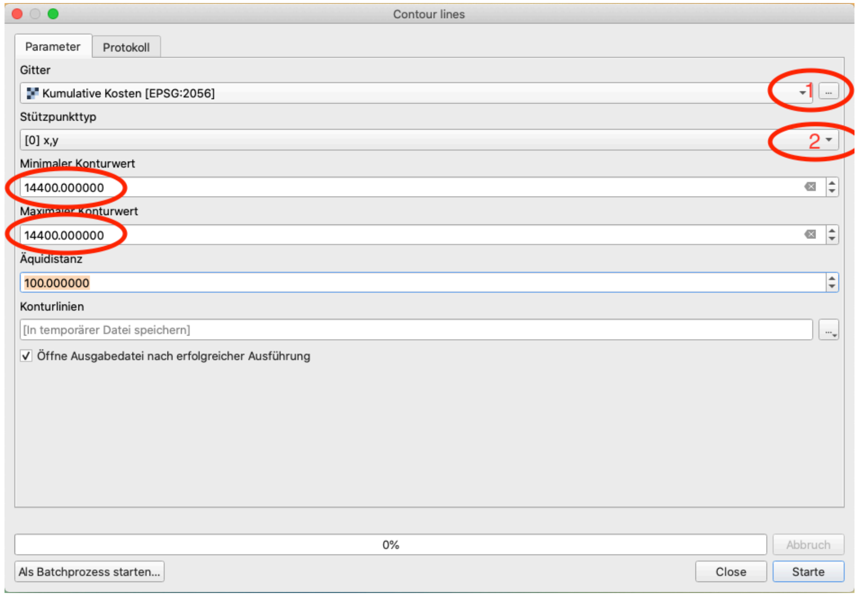
*left: movement directions; right: cumulative movement costs, colored with pseudocolor spectra, inverse*

# Calculate the movement distance within a given time

To get how far a pedestrian can walk in a given time, we can use the Contour tool, this time from SAGA:

- here we can set maximum and minimum value for the contour line
- the walking time is in seconds, so eg. 4 hours is equivalent with 4 x 60 x 60 = 14400 seconds

- search for 'contour' in the Toolbox ⚙
- Click on 'Contour lines' tool from SAGA
- Set 'Cumulative Costs' as layer
- Set x,y as 'Support point type'
- Set maximum and minimum to 14400
- click on 'Run'



Contour lines

Gitter: Kumulative Kosten [EPSG:2056]

Stützpunkttyp: [0] x,y

Minimaler Konturwert: 14400.000000

Maximaler Konturwert: 14400.000000

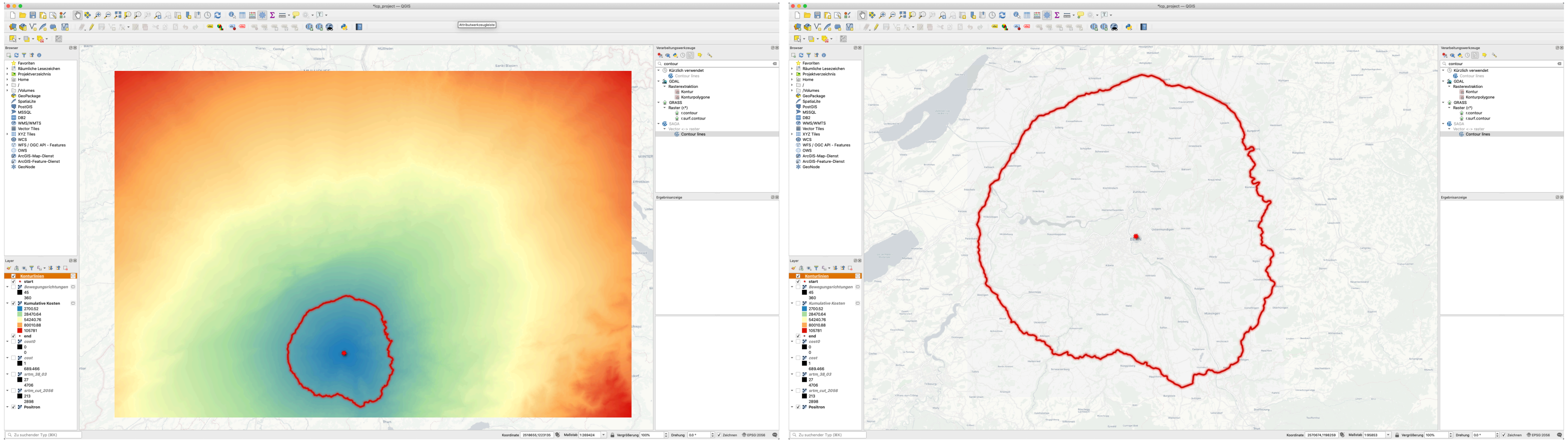
Äquidistanz: 100.000000

Konturlinien: [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

# Walking distance results

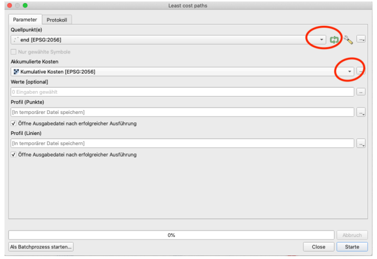
- You should get a line showing the extent of a walking time of 5h
- You can make it stick out more using Symbology (here: neon glow)
- If you make the actual cost layer invisible, you can compare with the base map
- In 4 hours I could get eg. to Kirchdorf
- Compare with Google Maps: between 3h 50 min and 4h 10 min
- or try it out yourself



*left: movement directions; right: cumulative movement costs, colored with pseudocolor spectra, inverse*

# Outlook

- You can use the results from r.walk also for Least Cost Path analysis:
  - Use the SAGA 'least cost paths' tool
  - Use the 'Cumulative costs' as input layer and the end (Basel) as source point layer
- take care for the selection of the right movement function and use kings move if possible
- you can combine the cumulative costs starting from both points combined to estimate movement corridors using the raster calculator and addition



Least cost paths

Parameter | Protokoll

Quellpunkt(e): end [EPSG:2056]

□ Nur gewählte Symbole

Akkumulierte Kosten: Kumulative Kosten [EPSG:2056]

Werte [optional]: 0 Eingaben gewählt

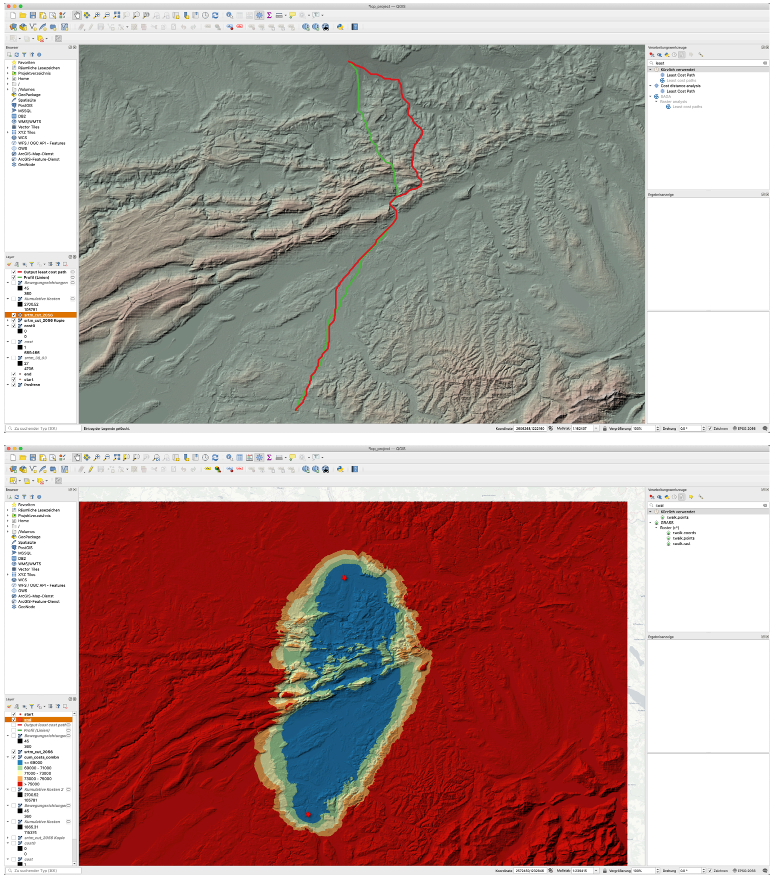
Profil (Punkte): [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

Profil (Linien): [In temporärer Datei speichern]

☑ Öffne Ausgabedatei nach erfolgreicher Ausführung

*calculating LCP with Cumulative Costs from r.walk and Least Cost Paths tool from SAGA*



*top: comparison between LCP from plugin (red) and from r.walk (green); bottom: Least Cost Corridor by combining Cumulative Costs starting from both points*

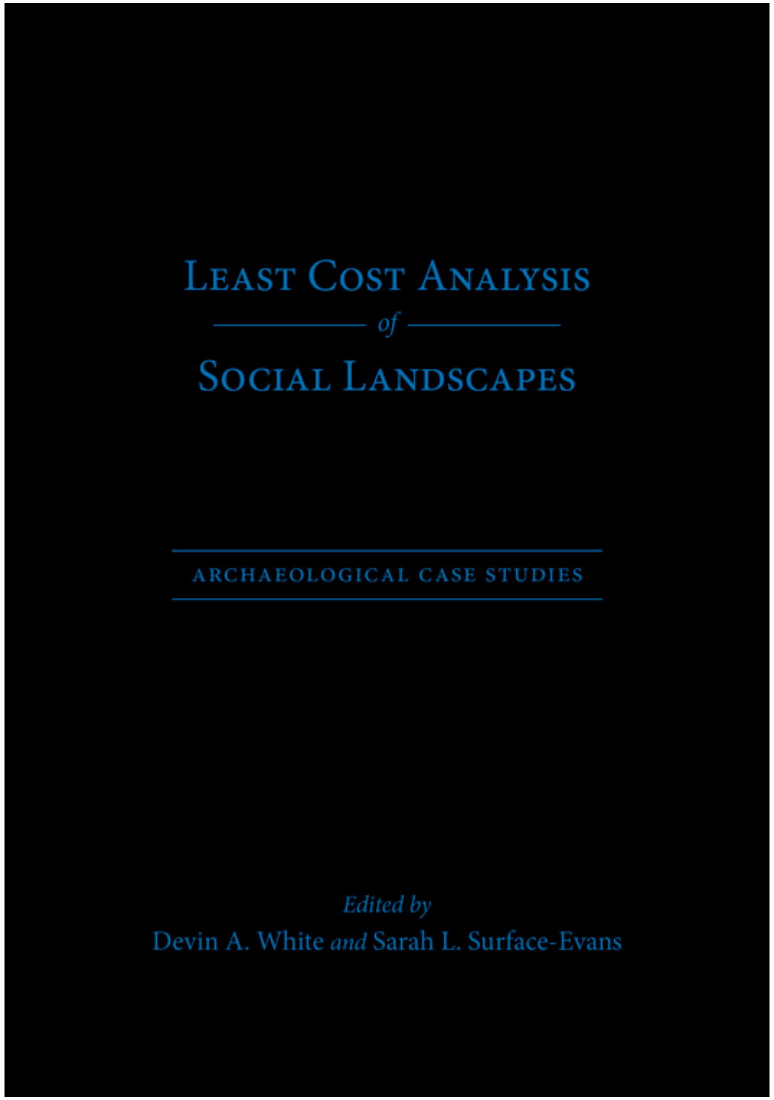
# What We've Covered

- Creating a cost layer from Slope
- Basic Least Cost Path calculation
- Creating a cumulative cost (walking time) layer
- Estimation of walking distance in a given time

# More on Least Cost Path Analysis

White, Devin A. and Sarah L Surface-Evans. Least Cost Analysis of Social Landscapes: Archaeological Case Studies. University of Utah Press, 2012. Project MUSE muse.jhu.edu/book/41407.

Free available from within the university network.


LEAST COST ANALYSIS of SOCIAL LANDSCAPES

ARCHAEOLOGICAL CASE STUDIES

*Edited by*
Devin A. White *and* Sarah L. Surface-Evans

# Homework

- Select two locations of your choice (in Switzerland or the world)
- Calculate the least cost path between them
- Send me a screenshot

# Any questions?

*Source: https://www.instagram.com/sadtopographies*

You might find the course material (including the presentations) at

https://github.com/BernCoDALab/gia

You can see the rendered presentations at

https://berncodalab.github.io/gia

You can contact me at

martin.hinz@unibe.ch