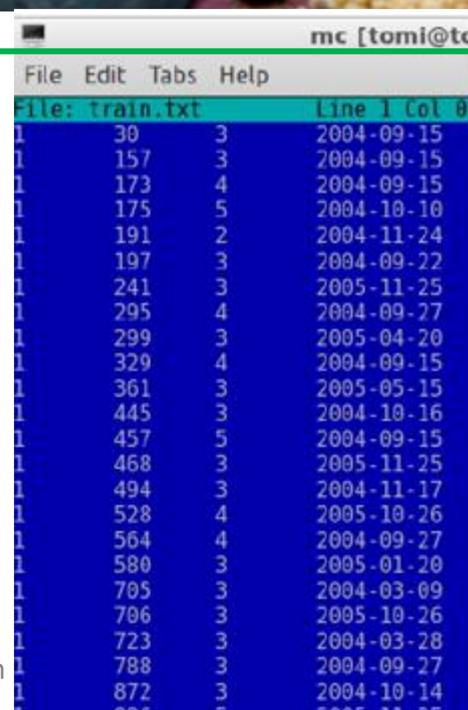
# NSWI166 Introduction to recommender systems and user preferences

Prostřední třetinu Peter Vojtáš, KSI MFF UK
první a poslední třetinu Láďa Peška, KSI MFF UK

7/12 Learn user preferences as a FLN-LMPM
(from Fagin-Lotem-Naor class of models)

# Old system

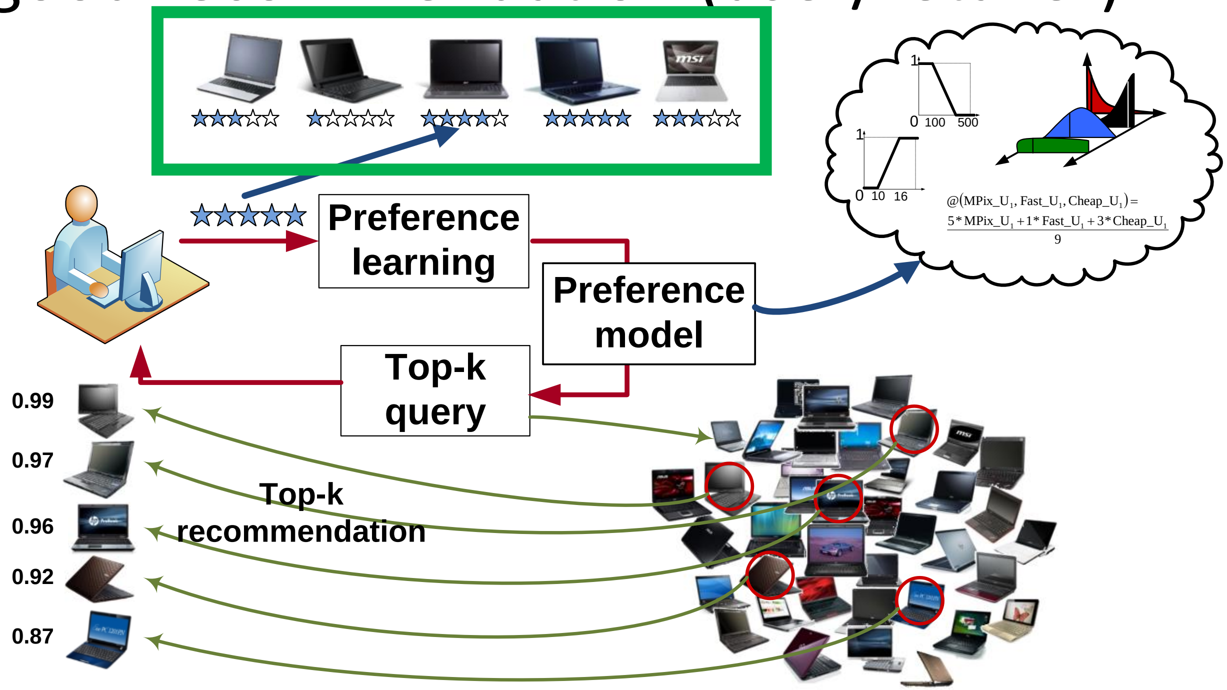

# new one - streaming


Netflix competition

# User’s preference learning

- Preference learning = generalization of our **observation** of user = **estimation** of his/her future acts – what is a good recommendation (user/retailer)
- We have
  - User behavior
- Would like to have an
  - LMPM user model
  - to compute top-k for recommendation
- What is our goal?



# Note the difference: induction/deduction + test

- Deduction = querying, search, input offline ordered

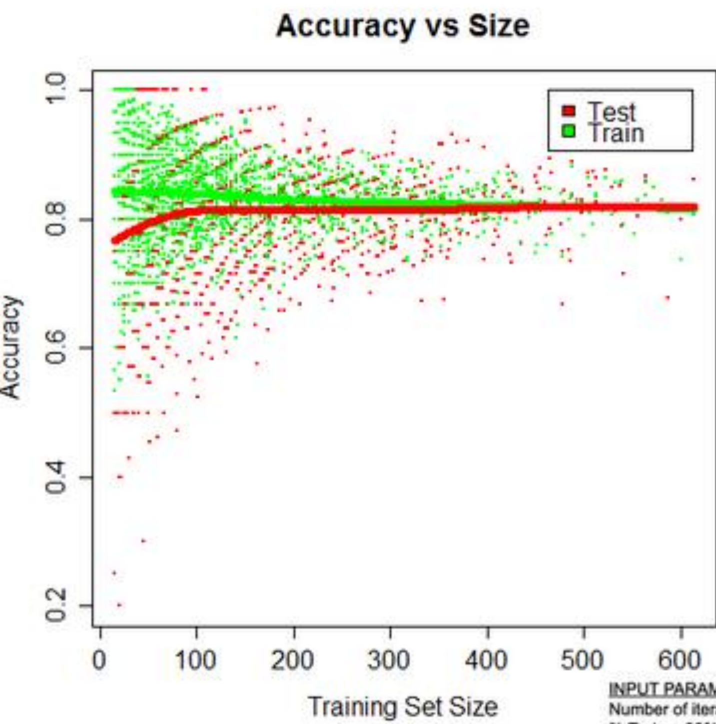
| P | 100m | P | Long | P | Shot | P | High | P | 400m | P | 110mh | P | Discus | P | Pole | P | Javelin | P | 1500r | PRAH |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 942 | 1 | 1089 | 4 | 847 | 1 | 915 | 2 | 964 | 1 | 985 | 4 | 840 | 2 | 1004 | 1 | 892 | 5 | 799 | 9277 |
| 4 | 938 | 2 | 1010 | 3 | 841 | 5 | 915 | 8 | 924 | 3 | 976 | 1 | 827 | 4 | 972 | 6 | 861 | 1 | 798 | 9062 |
| 2 | 922 | 3 | 982 | 9 | 831 | 7 | 859 | 1 | 919 | 4 | 946 | 3 | 803 | 11 | 941 | 2 | 843 | 12 | 770 | 8816 |
| 17 | 915 | 8 | 932 | 1 | 810 | 12 | 831 | 9 | 909 | 7 | 936 | 5 | 803 | 10 | 910 | 3 | 839 | 10 | 760 | 8645 |
| 3 | 897 | 6 | 908 | 8 | 800 | 4 | 803 | 14 | 877 | 10 | 936 | 7 | 800 | 12 | 910 | 14 | 797 | 11 | 734 | 8462 |
| 10 | 890 | 11 | 898 | 16 | 796 | 13 | 803 | 3 | 873 | 9 | 929 | 9 | 796 | 9 | 880 | 15 | 763 | 3 | 721 | 8349 |
| 14 | 885 | 4 | 891 | 10 | 780 | 2 | 776 | 17 | 873 | 2 | 916 | 11 | 748 | 1 | 849 | 5 | 746 | 4 | 706 | 8170 |
| 8 | 883 | 5 | 859 | 7 | 776 | 3 | 776 | 10 | 872 | 8 | 913 | 2 | 732 | 6 | 849 | 16 | 737 | 6 | 703 | 8100 |
| 6 | 876 | 12 | 854 | 6 | 772 | 14 | 776 | 5 | 870 | 6 | 903 | 8 | 698 | 8 | 849 | 7 | 735 | 2 | 686 | 8019 |
| 9 | 863 | 7 | 853 | 17 | 769 | 15 | 776 | 4 | 866 | 12 | 897 | 12 | 696 | 3 | 819 | 10 | 715 | 16 | 679 | 7933 |
| 13 | 863 | 9 | 840 | 5 | 765 | 6 | 749 | 7 | 858 | 14 | 886 | 15 | 691 | 7 | 819 | 17 | 711 | 8 | 665 | 7847 |
| 5 | 858 | 13 | 840 | 2 | 751 | 8 | 749 | 13 | 849 | 15 | 870 | 14 | 688 | 15 | 790 | 11 | 709 | 9 | 664 | 7768 |
| 16 | 854 | 10 | 799 | 11 | 739 | 16 | 749 | 6 | 846 | 17 | 853 | 10 | 672 | 5 | 760 | 8 | 672 | 13 | 640 | 7584 |
| 7 | 843 | 15 | 797 | 13 | 715 | 9 | 723 | 16 | 819 | 11 | 842 | 13 | 668 | 13 | 760 | 4 | 656 | 15 | 636 | 7459 |
| 12 | 841 | 14 | 788 | 14 | 708 | 11 | 696 | 12 | 808 | 13 | 841 | 6 | 655 | 17 | 731 | 13 | 653 | 17 | 628 | 7349 |
| 11 | 793 | 17 | 774 | 12 | 667 | 10 | 670 | 15 | 803 | 16 | 817 | 16 | 653 | 16 | 673 | 12 | 617 | 7 | 621 | 7088 |
| 15 | 784 | 16 | 769 | 15 | 666 | 17 | 644 | 11 | 791 | 5 | 798 | 17 | 608 | 14 | 645 | 9 | 593 | 14 | 563 | 6861 |

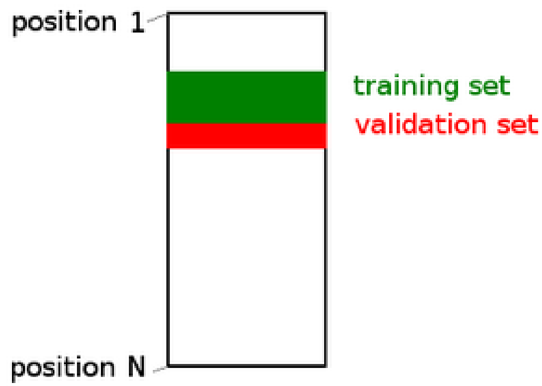
| P | Athlete | Points | |
|---|---|---|---|
| 1 | Šebrle CZE | 9026 | 3 |
| 2 | Nool EST | 8604 | 5 |
| 3 | Dvorak CZE | 8527 | 5 |
| 4 | Lobodin RUS | 8465 | 5 |
| 5 | Zsivoczky HUN | 8173 | 7 |
| 6 | Ambrosch AUT | 8122 | 8 |
| 7 | Kürtösi HUN | 8099 | 9 |
| 8 | Warners NED | 8085 | 9 |
| 9 | Hämäläinen FIN | 8028 | 9 |
| 10 | Jensen NOR | 8004 | 10 |
| 11 | Schönbeck GEI | 7891 | 11 |
| 12 | Niklaus GER | 7891 | 11 |
| 13 | Tebbich AUT | 7632 | 13 |
| 14 | Llanos PUR | 7613 | 13 |
| 15 | SchnallingerAU | 7576 | 14 |
| 16 | Walser AUT | 7546 | 14 |
| 17 | Walser AUT | 7506 | 14 |

- Induction = learning, estimating, generalizing, …

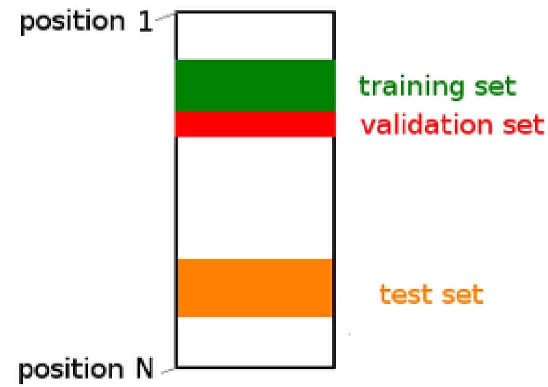
| P=ID | Athlete | 100m | Long | Shot | High | 400m | 110mh | Discus | Pole | Javelin | 1500m | Points | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Šebrle CZE | 10,64 | 8.11 | 15.33 | 2.12 | 47,79 | 13,92 | 47.92 | 4.8 | 70.16 | 4.21,98 | 9026.00 | train set |
| 2 | Nool EST | 10,73 | 7.8 | 14.37 | 1.97 | 46,89 | 14,46 | 43.32 | 5.3 | 66.94 | 4.39,11 | 8604.00 | train set |
| 3 | Dvorak CZE | 10,84 | 7.69 | 15.83 | 1.97 | 48,76 | 13,99 | 46.74 | 4.7 | 66.66 | 4.33,58 | 8527.00 | train set |
| 4 | Lobodin RUS | 10,66 | 7.32 | 15.93 | 2 | 48,91 | 14,22 | 48.53 | 5.2 | 54.56 | 4.35,97 | 8465.00 | train set |
| 5 | Zsivoczky HUN | hidden for the learning algorithm - not in train set - left for test set | | | | | | | | | | | |
| 6 | Ambrosch AUT | 10,93 | 7.39 | 14.71 | 1.94 | 49,33 | 14,56 | 39.52 | 4.8 | 68.15 | 4.36,36 | 8122.00 | train set |
| 7 | Kürtösi HUN | 11,08 | 7.16 | 14.77 | 2.06 | 49,07 | 14,30 | 46.61 | 4.7 | 59.83 | 4.49,58 | 8099.00 | train set |
| 8 | Warners NED | 10,90 | 7.49 | 15.16 | 1.94 | 47,70 | 14,48 | 41.64 | 4.8 | 55.62 | 4.42,47 | 8085.00 | train set |
| 9 | Hämäläinen FIN | 10,99 | 7.11 | 15.67 | 1.91 | 48,01 | 14,36 | 46.41 | 4.9 | 50.33 | 4.42,66 | 8028.00 | train set |
| 10 | Jensen NOR | 10,87 | 6.94 | 14.85 | 1.85 | 48,77 | 14,30 | 40.38 | 5 | 58.51 | 4.27,65 | 8004.00 | train set |
| 11 | Schönbeck GER | 11,31 | 7.35 | 14.17 | 1.88 | 50,51 | 15,06 | 44.07 | 5.1 | 58.11 | 4.31,69 | 7891.00 | train set |
| 12 | Niklaus GER | 11,09 | 7.17 | 12.99 | 2.03 | 50,14 | 14,61 | 41.56 | 5 | 51.95 | 4.26,13 | 7891.00 | train set |
| 13 | Tebbich AUT | hidden for the learning algorithm - not in train set - left for test set | | | | | | | | | | | |
| 14 | Llanos PUR | 10,89 | 6.89 | 13.67 | 1.97 | 48,67 | 14,70 | 41.14 | 4.1 | 63.93 | 4.59,38 | 7613.00 | train set |
| 15 | SchnallingerAUT | 11,35 | 6.93 | 12.98 | 1.97 | 50,25 | 14,83 | 41.3 | 4.6 | 61.65 | 4.47,22 | 7576.00 | train set |
| 16 | Walser AUT | 11,03 | 6.81 | 15.1 | 1.94 | 49,90 | 15,27 | 39.45 | 4.2 | 59.97 | 4.40,22 | 7546.00 | train set |
| 17 | Walser AUT | 10,76 | 6.83 | 14.67 | 1.82 | 48,76 | 14,97 | 37.2 | 4.4 | 58.23 | 4.48,52 | 7506.00 | train set |
| | | | | | | | | | | | | | |
| | Zsivoczky HUN | 11,01 | 7,19 | 14,6 | 2,12 | 48,81 | 15,43 | 46,73 | 4,5 | 60,57 | 4.21,85 | | data |
| | f, t model | f1(11.01) | f2(7.19) | f3(14.6) | f4(2.12) | f5(48.81) | f6(15.43) | f7(46.73) | f8(4.5) | f9(60.57) | f10(4.21,85) | t(C22, …,L22) | estimation |
| | Zsivoczky HUN | 858 | 859 | 765 | 915 | 870 | 798 | 803 | 760 | 746 | 799 | 8173.00 | test set |
| | | | | | | | | | | | | | |
| | Tebbich AUT | 10,99 | 7.11 | 13.78 | 2 | 49,26 | 15,07 | 40.18 | 4.5 | 54.32 | 4.46,57 | | data |
| | | f1(10.99) | f2(7.11) | f3(13.78) | f4(2) | f5(49.26) | f6(15.07) | f7(40.18) | f8(4.5) | f9(54.32) | f10(4.46.57) | t(C26,…,L26) | estimation |
| | Tebbich AUT | 863 | 840 | 715 | 803 | 849 | 841 | 668 | 760 | 653 | 640 | 7632.00 | test set |

# Learning from train data, testing (Google images)



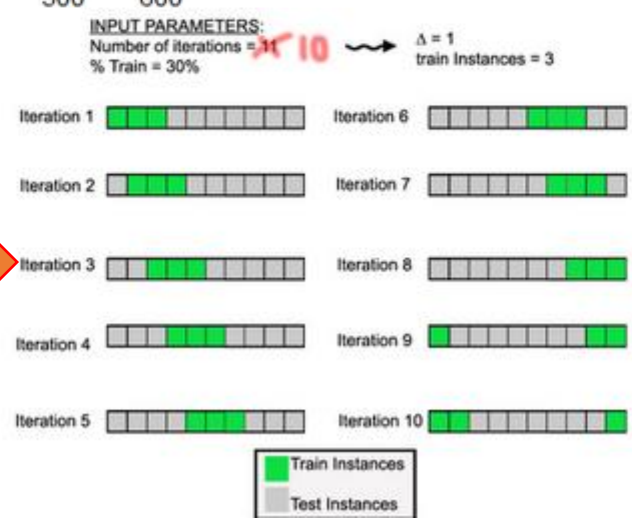



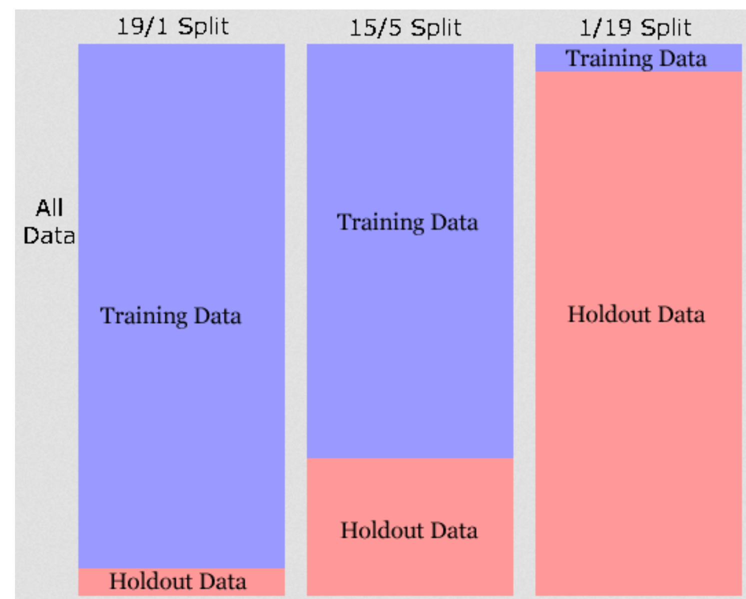

contiguous data sets:
on the validation set, my algorithm gets excellent results
MCC >= 0.9



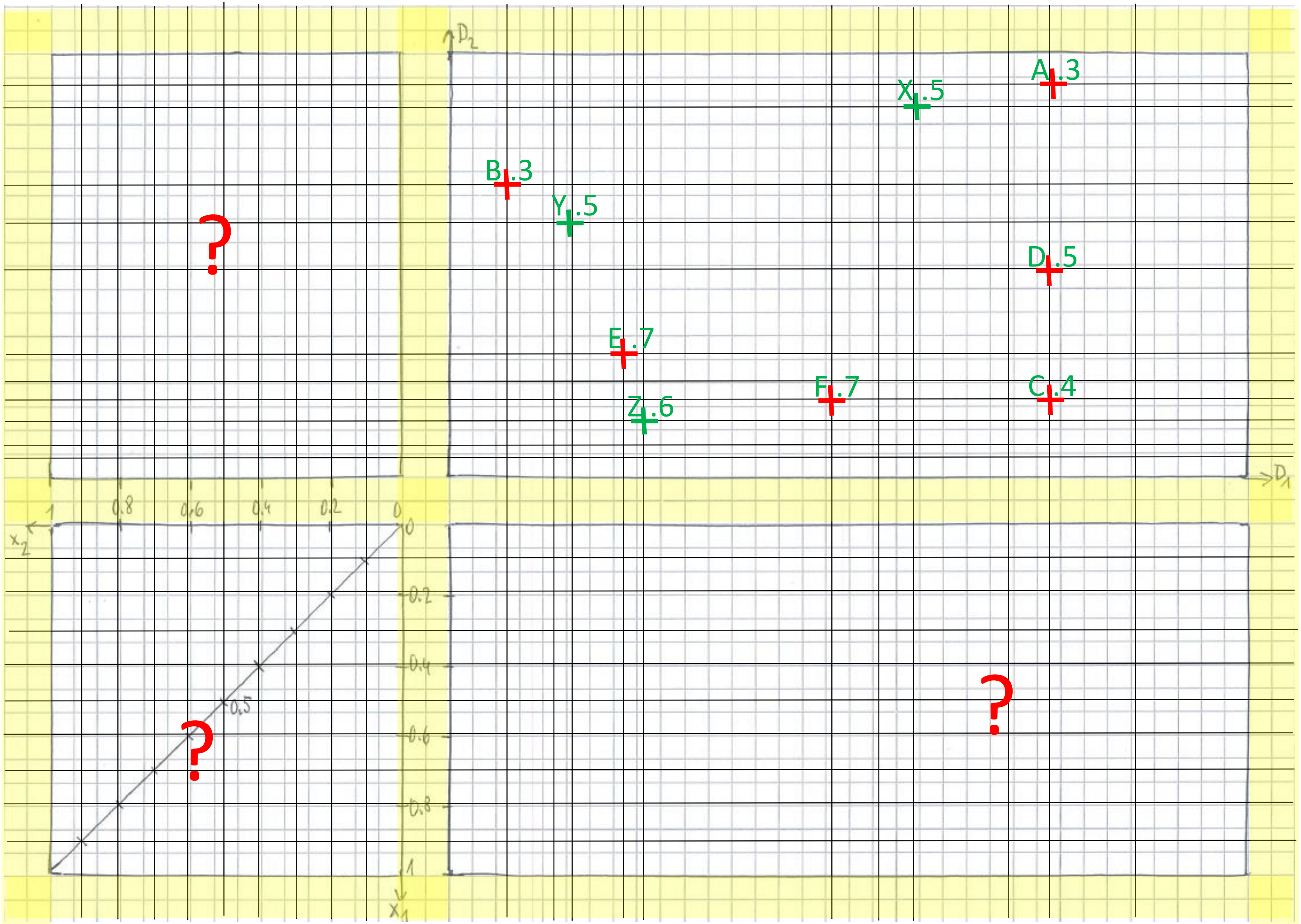

distant data sets:
on the test set, my algorithm usually gets bad results
MCC ~= +0.1

What is the conclusion?



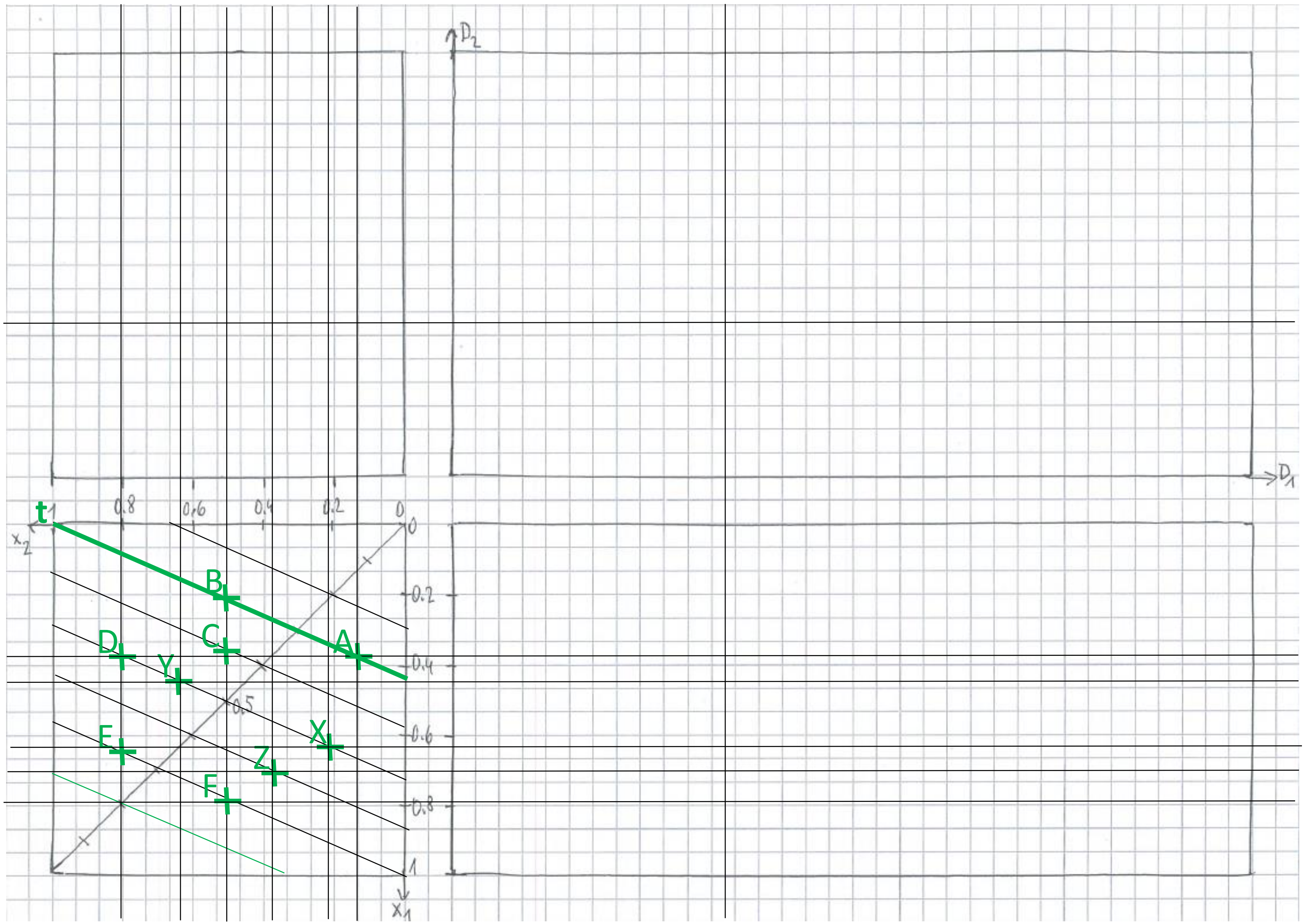
In the beginning we know only overall preferences $r^u$



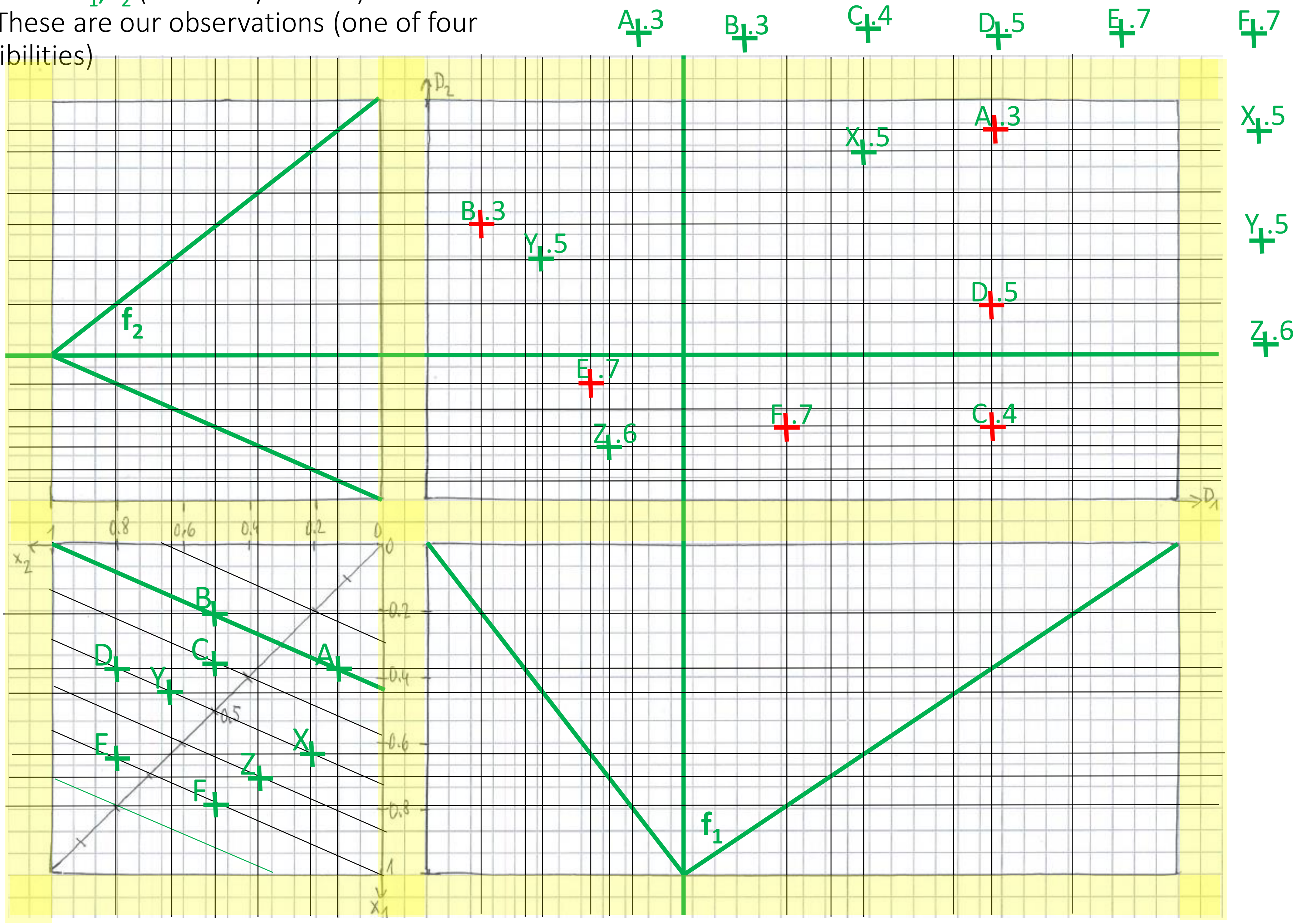
# Learning the LMPM model

- from data generated by an LMPM model
- so, we know for sure data are from an LMPM model
- We know that shape of attribute preferences are “hill”
- We chose aggregation contour line
- We start generating data from PC, in order to have “good” numeric values of overall preference – points are on contour lines with decimal values
- Selecting attribute preferences, we have starting data

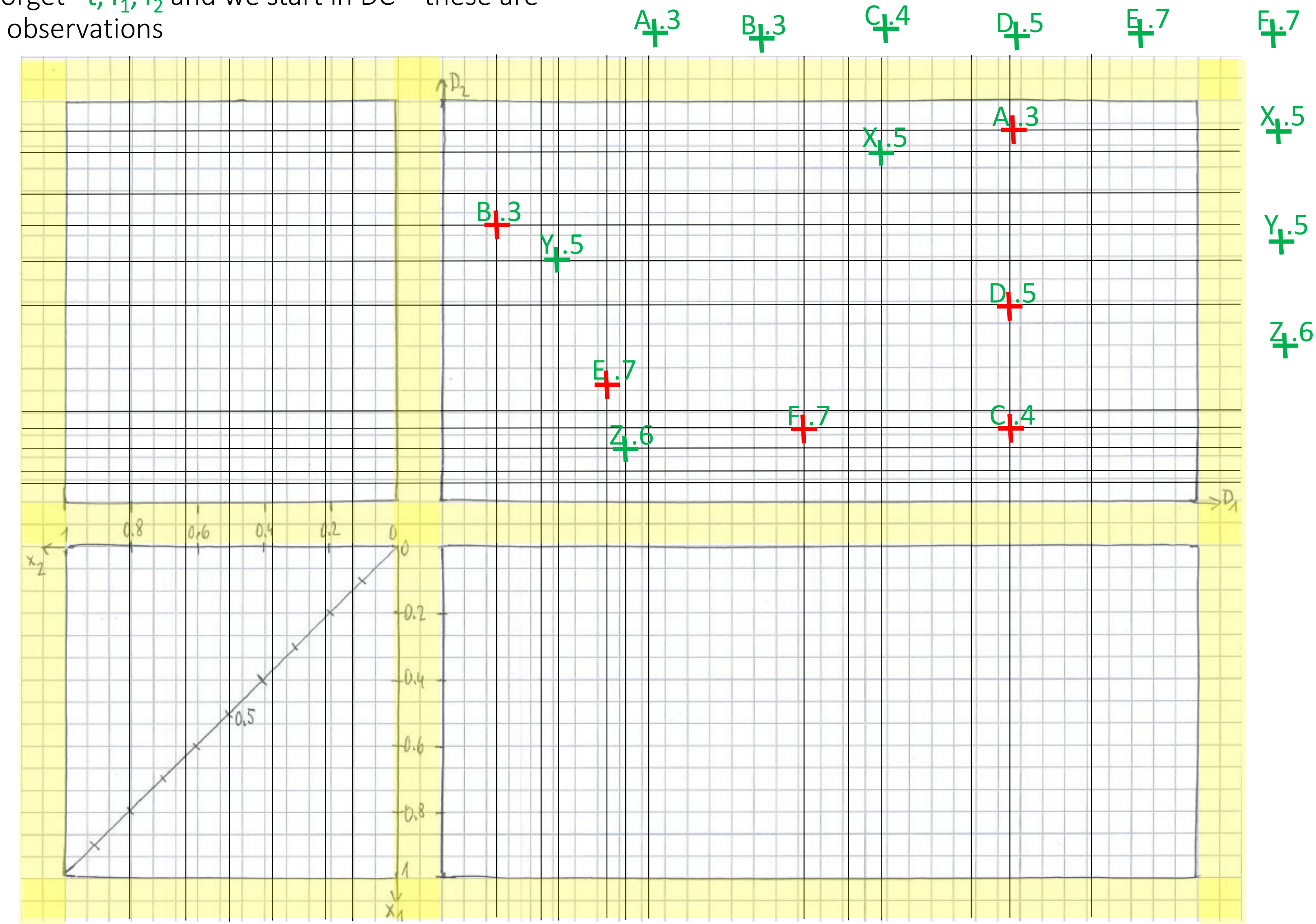
1. generate PC points, several in same contour line (**t** can be chosen arbitrarily) +
use squared underlining to by able to depict $r^u$ , $r^{ft}$ and after learning $r^u$ , $r^{ft}$ ,



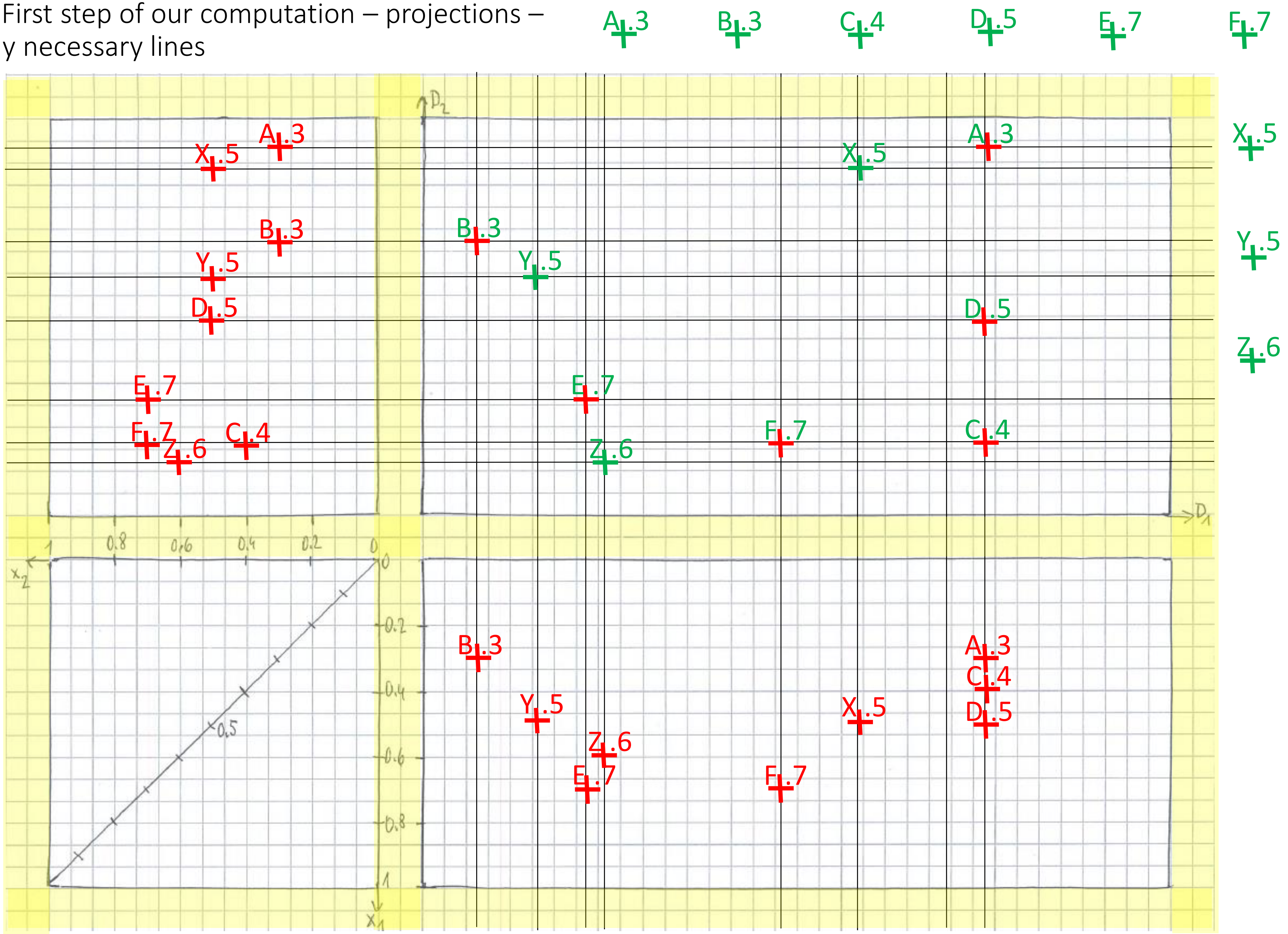
1. Choose $f_1$, $f_2$ (arbitrarily hill like) the inverse in DC. These are our observations (one of four possibilities)



2. Forget t, $f_1$, $f_2$ and we start in DC – these are our observations



# 3. 3. First step of our computation – projections – – only necessary lines



# Learning the LMPM model – from data generated by an LMPM model

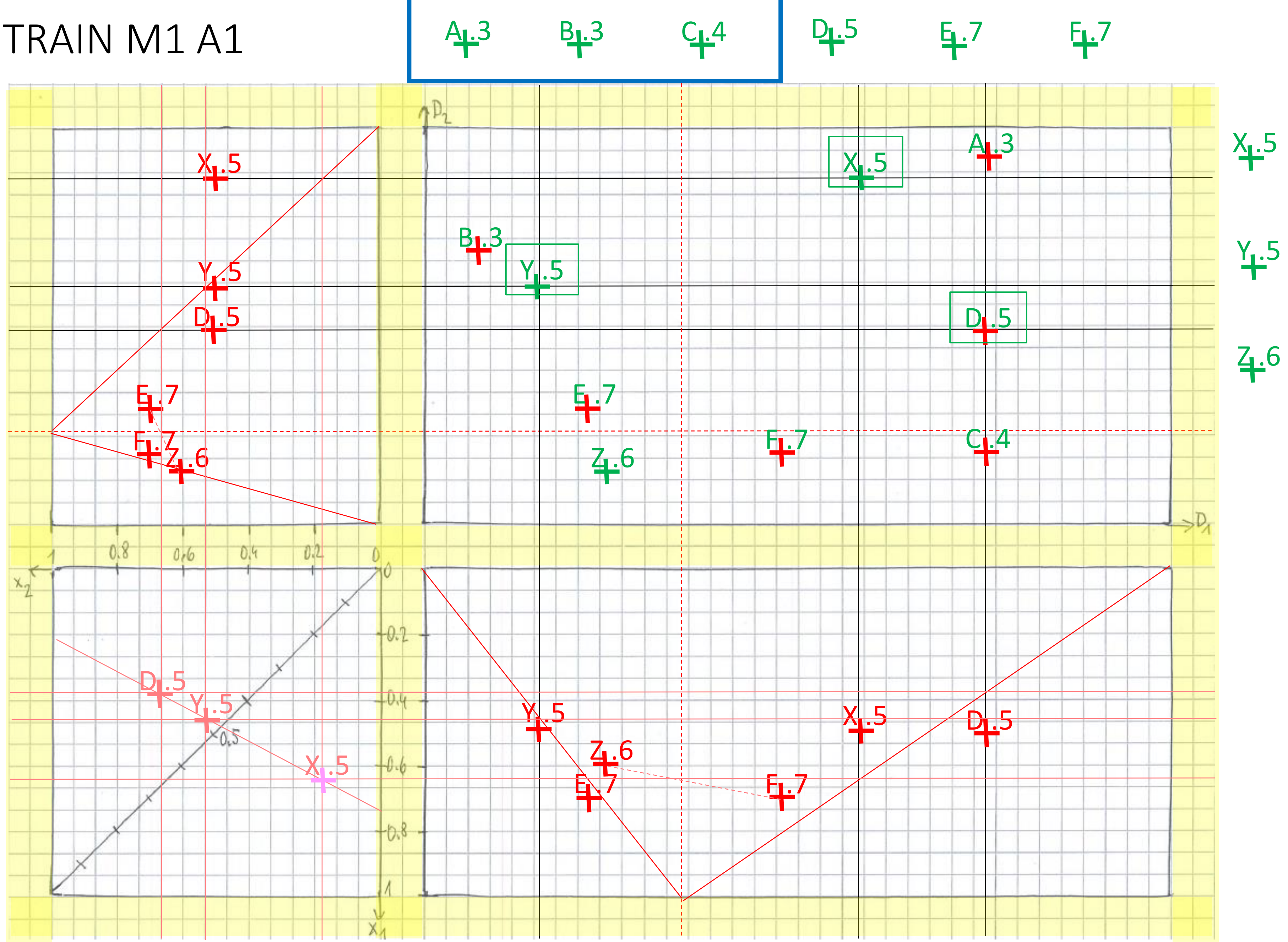
- We have single user preference data generated from a LMPM model (given $f_1$, $f_2$, $t$ and calculated $r^{ft}$)
- Learning starts projecting of points in DC to preference space $[0,1]xD_i$
- To learn $f_1$, $f_2$, we will specify two learning methods
  - **$m_1$** takes 2nd and 3rd biggest values and their center of mass is the estimated ideal point in the respective domain (ties case by case)
  - **$m_2$** tries to separate about half of data from 0 by lines from max/min at zero, their intersection domain coordinate is the estimated ideal point
- To learn $t$ we use the fact that in training there are always two pairs of items with same preference degree
  - **$a_1$** takes the pair with smaller preference and using $f_1$, $f_2$ maps them to PC and we expect that they define estimated contour line
  - **$a_2$** method does the same the pair with bigger preference
- So, there are 4 methods in total.

# Learning the LMPM model

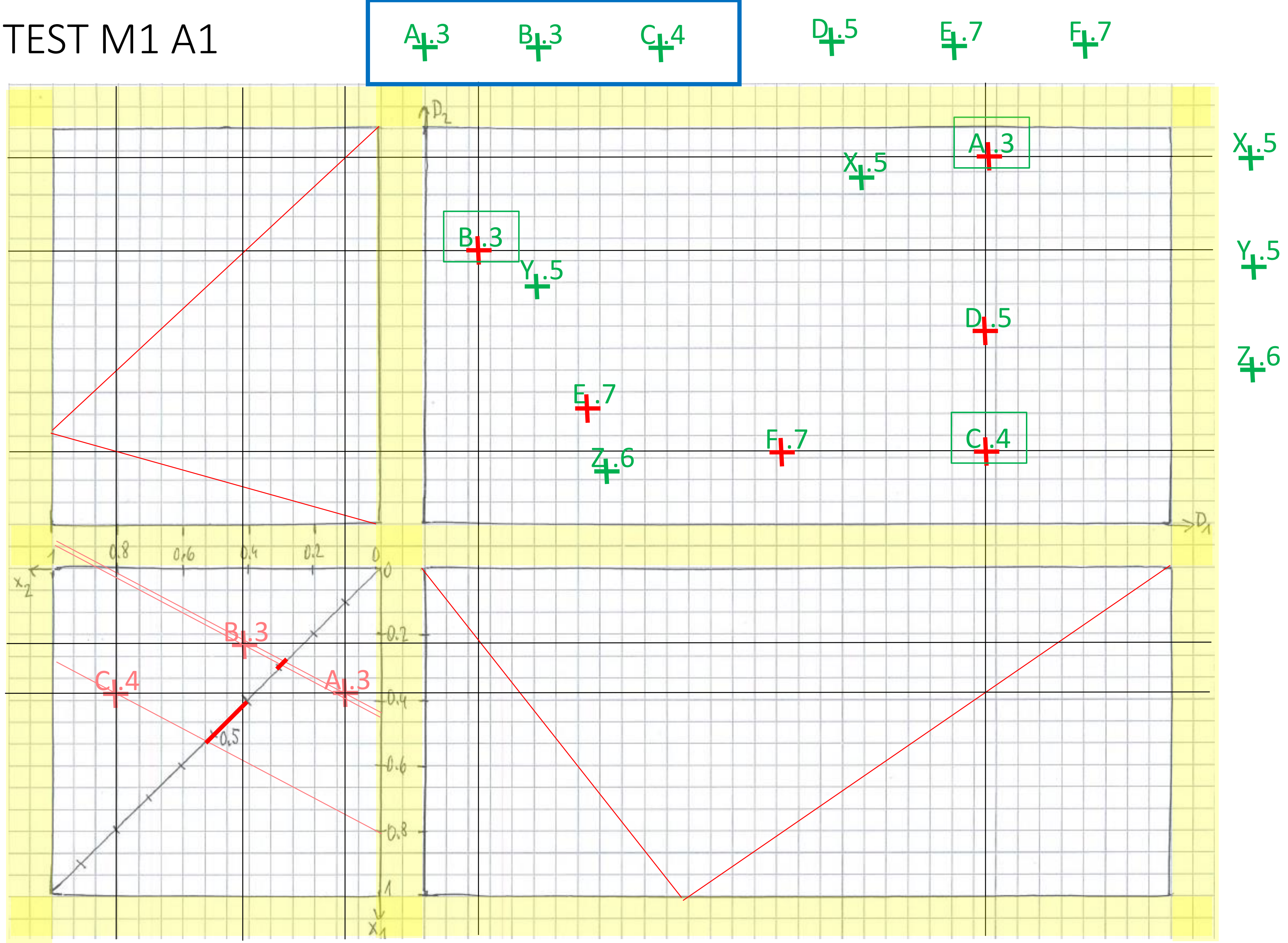
- For each $m_i a_j$ methods we use a 1/3 split of data to training and testing in $c_1$, $c_2$, $c_3$ iterations of learning – hence there will be in total 12 experiments $m_i a_j c_k$ calculating $r^{ft}$.
- Each of these 12 experiments evaluates error measure on test set. Metric used is sum of absolute values of differences between $r^{ft}$ and $r^{ft}$ degree on test set. So, we have 24 graphical "calculations"
- Better is the method which has smaller sum of split errors (and consequently smaller average of errors)

# C1 split

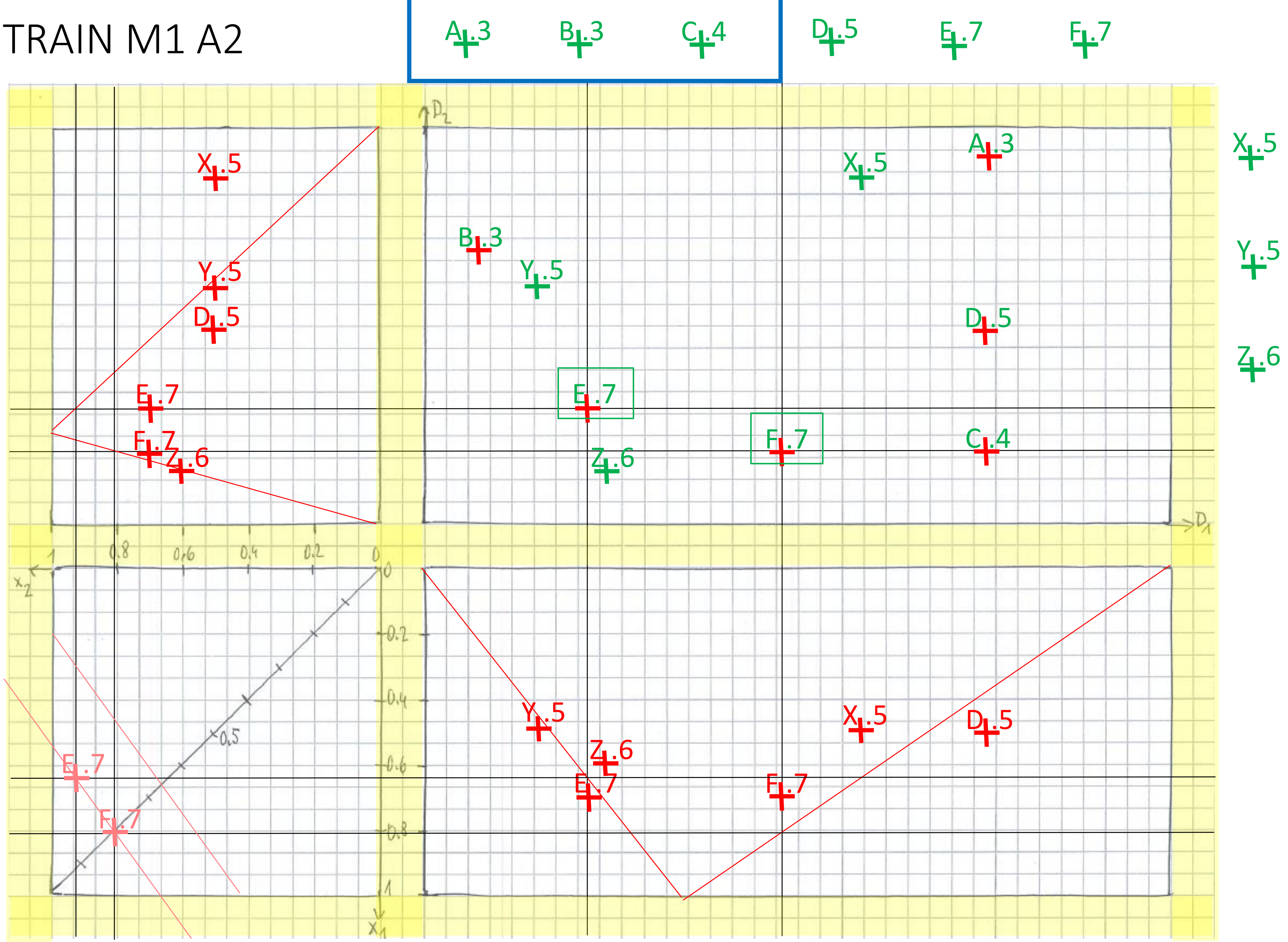
# C1 TRAIN M1 A1



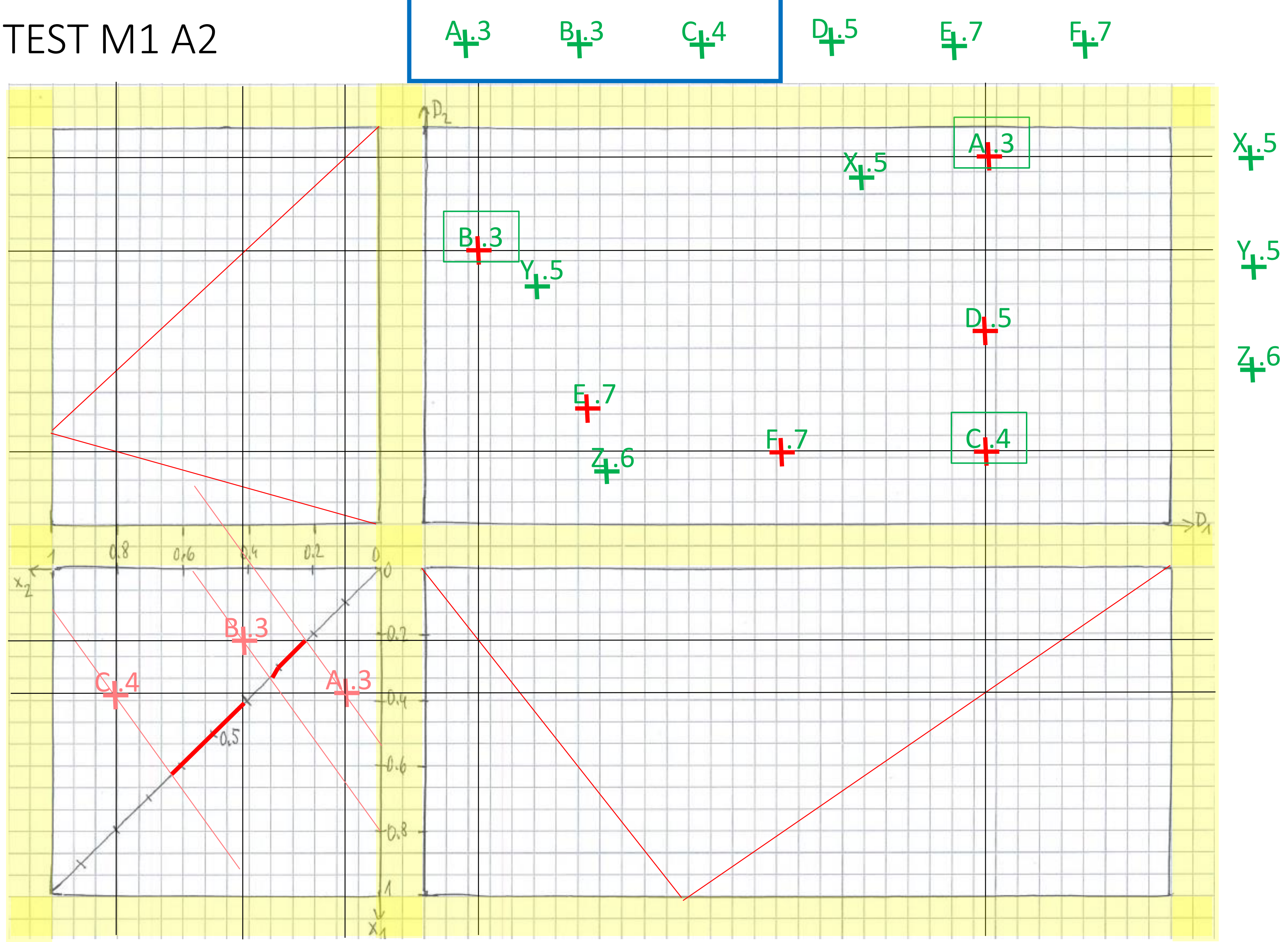
# C1 TEST M1 A1



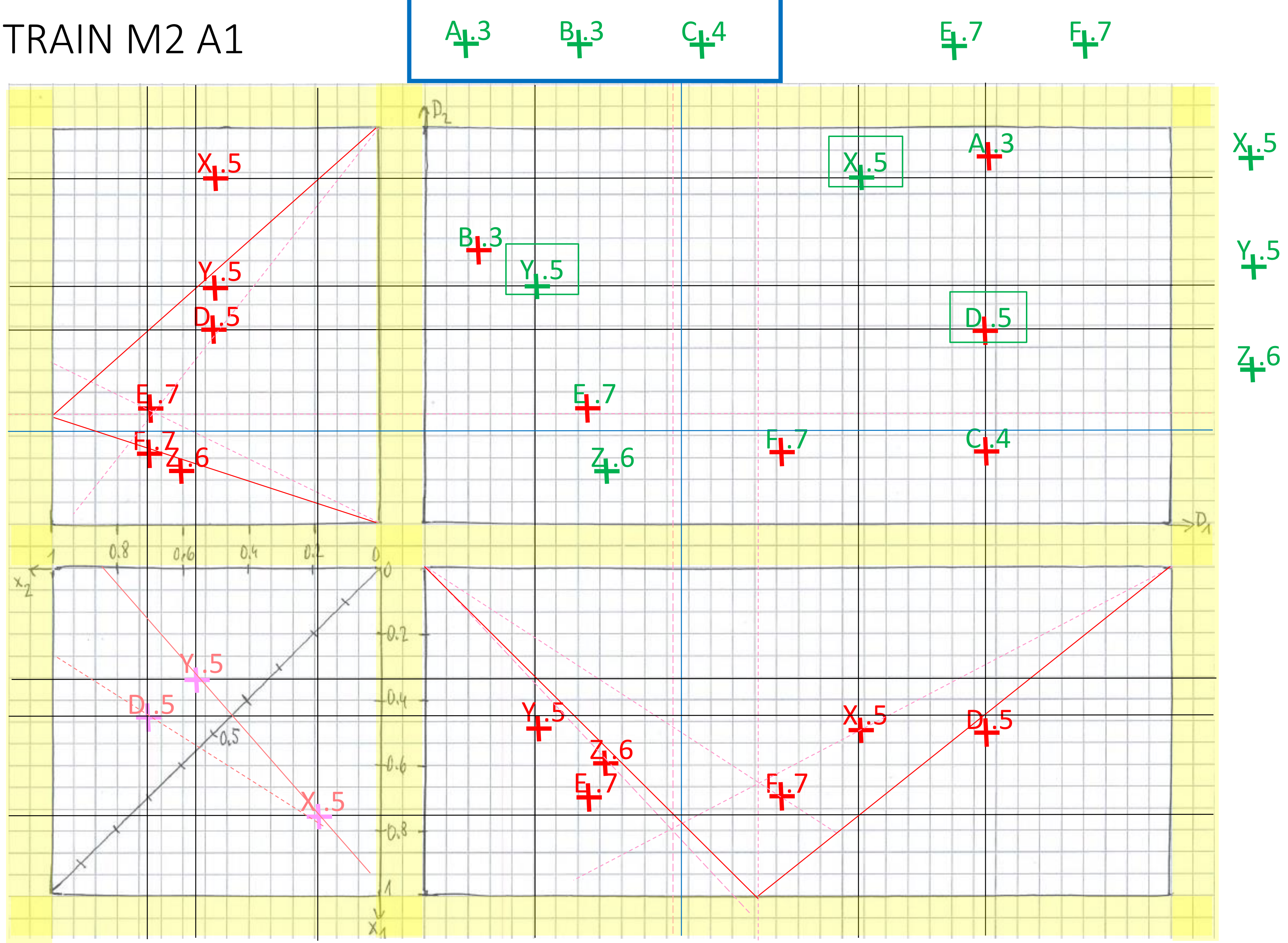
# C1 TRAIN M1 A2



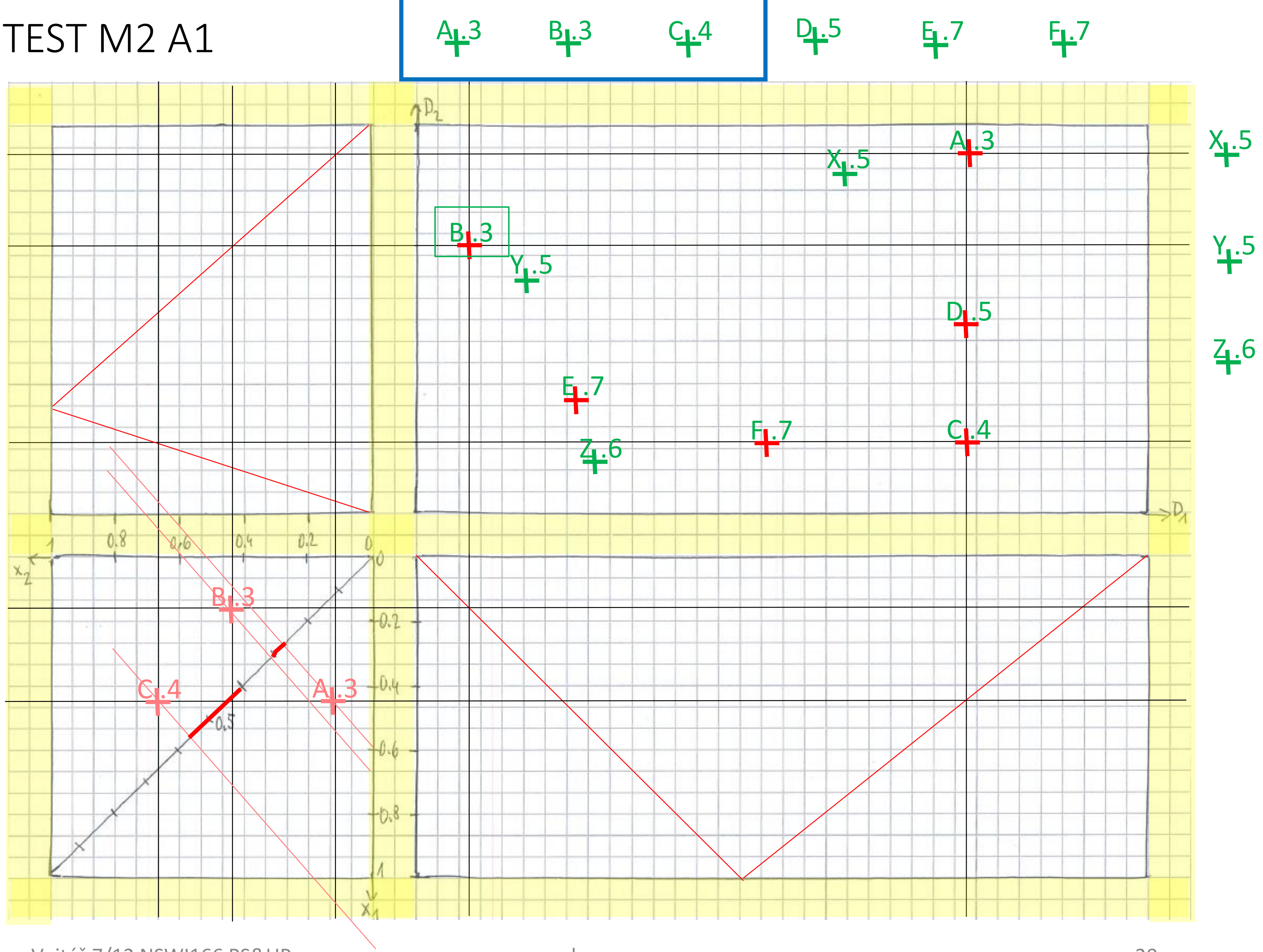
# C1 TEST M1 A2



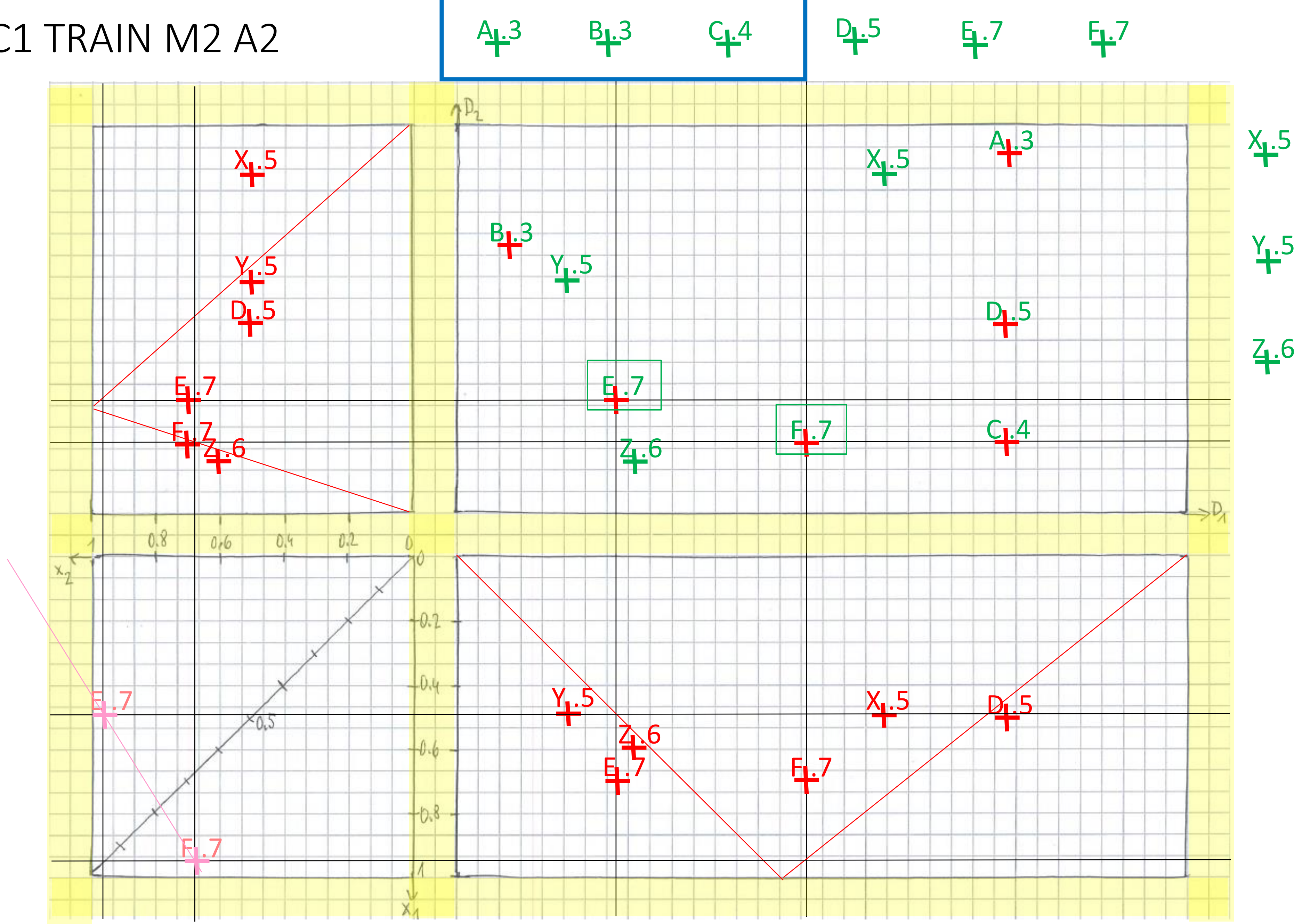
# C1 TRAIN M2 A1



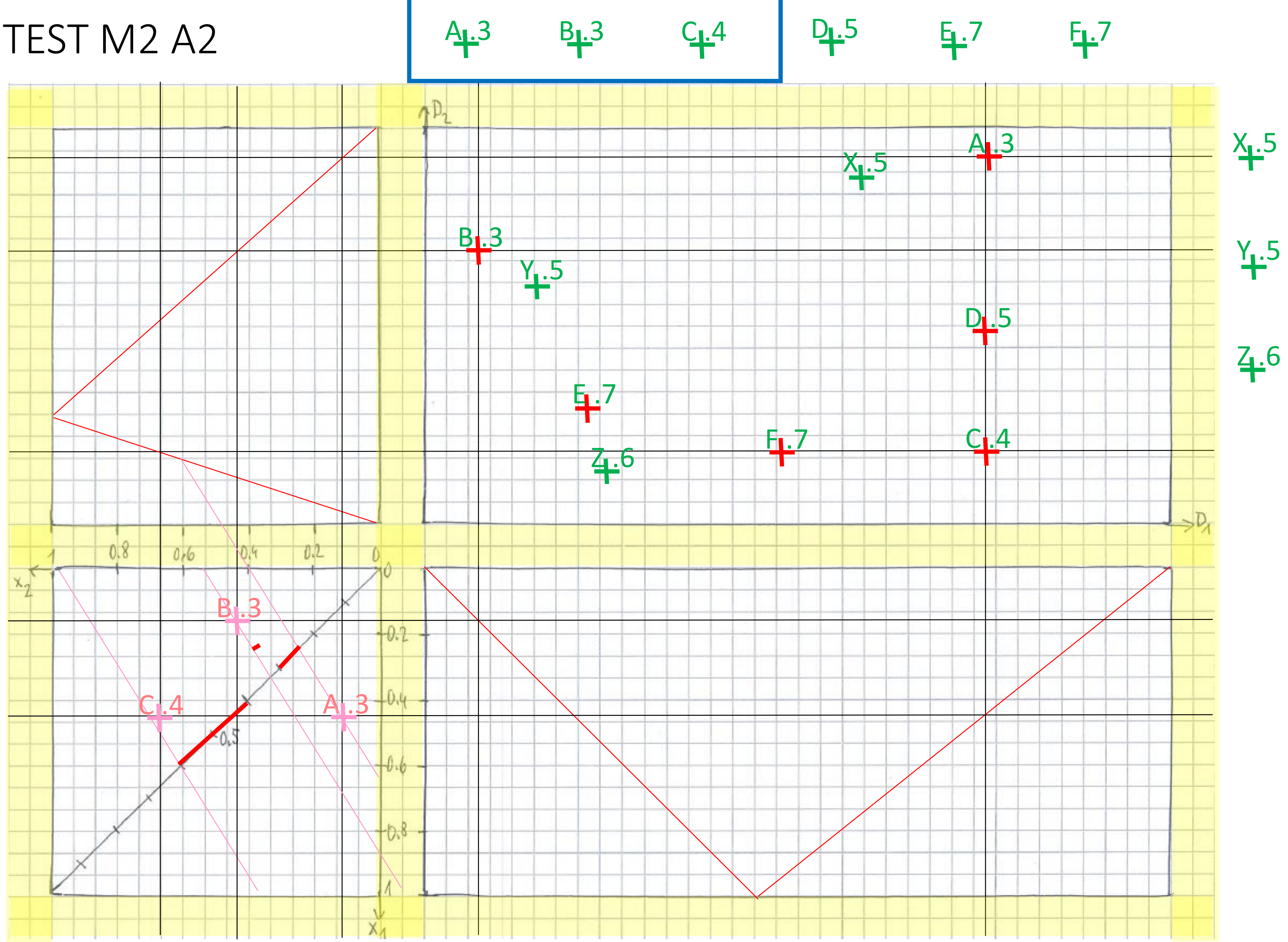
# C1 TEST M2 A1



# C1 TRAIN M2 A2

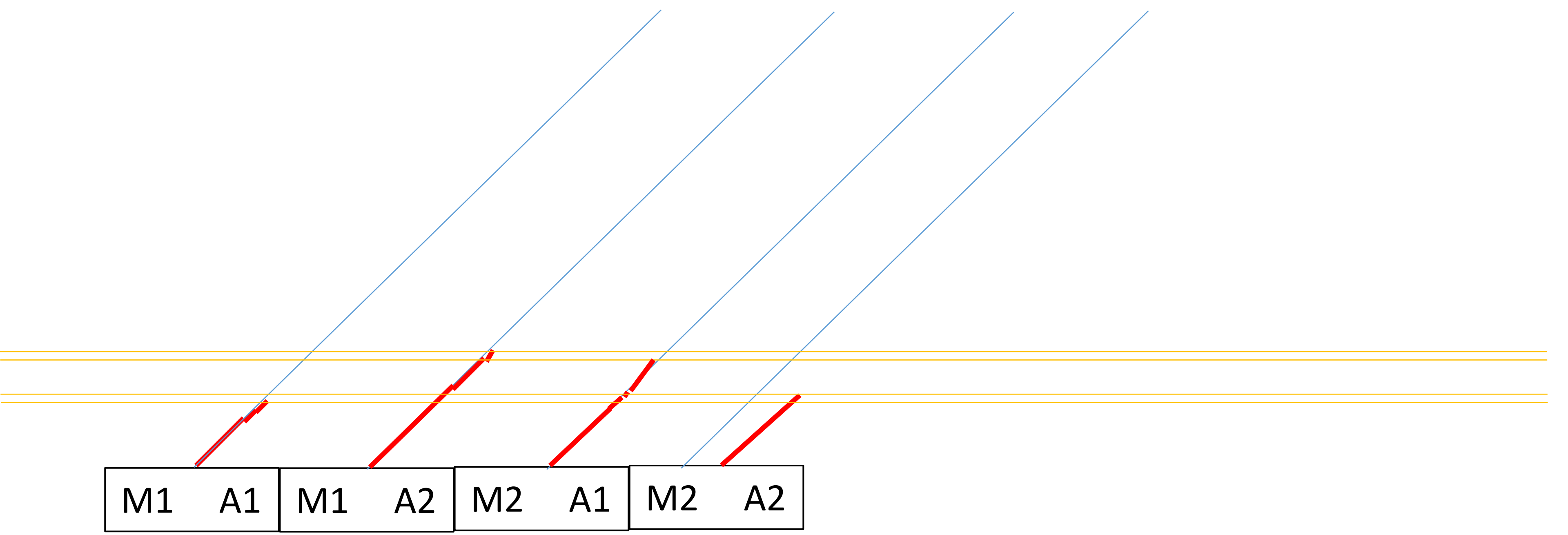


# C1 TEST M2 A2



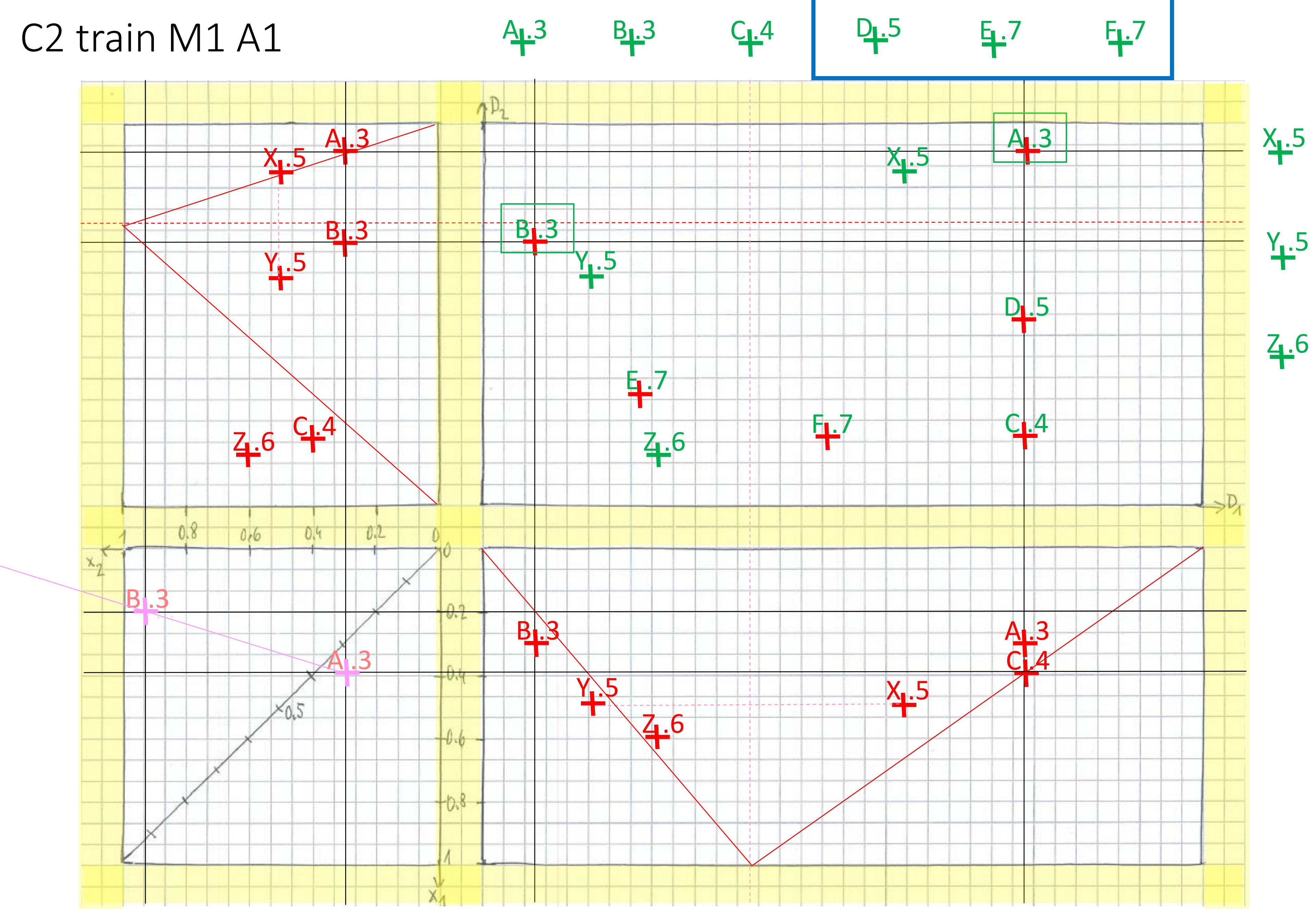
# Experiment evaluation

sum of error measures (3 points in test set) of split c1,

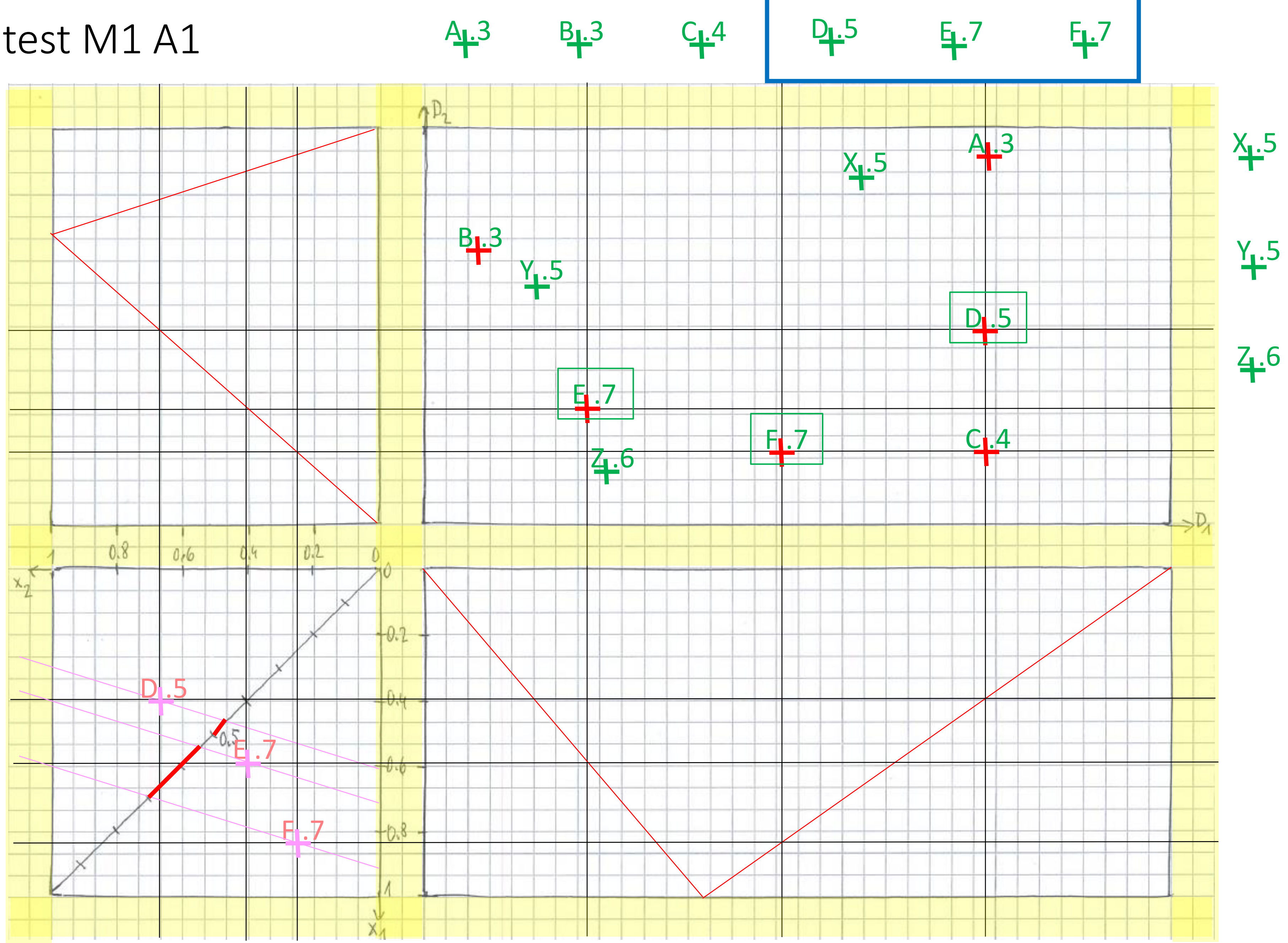


# C2 split

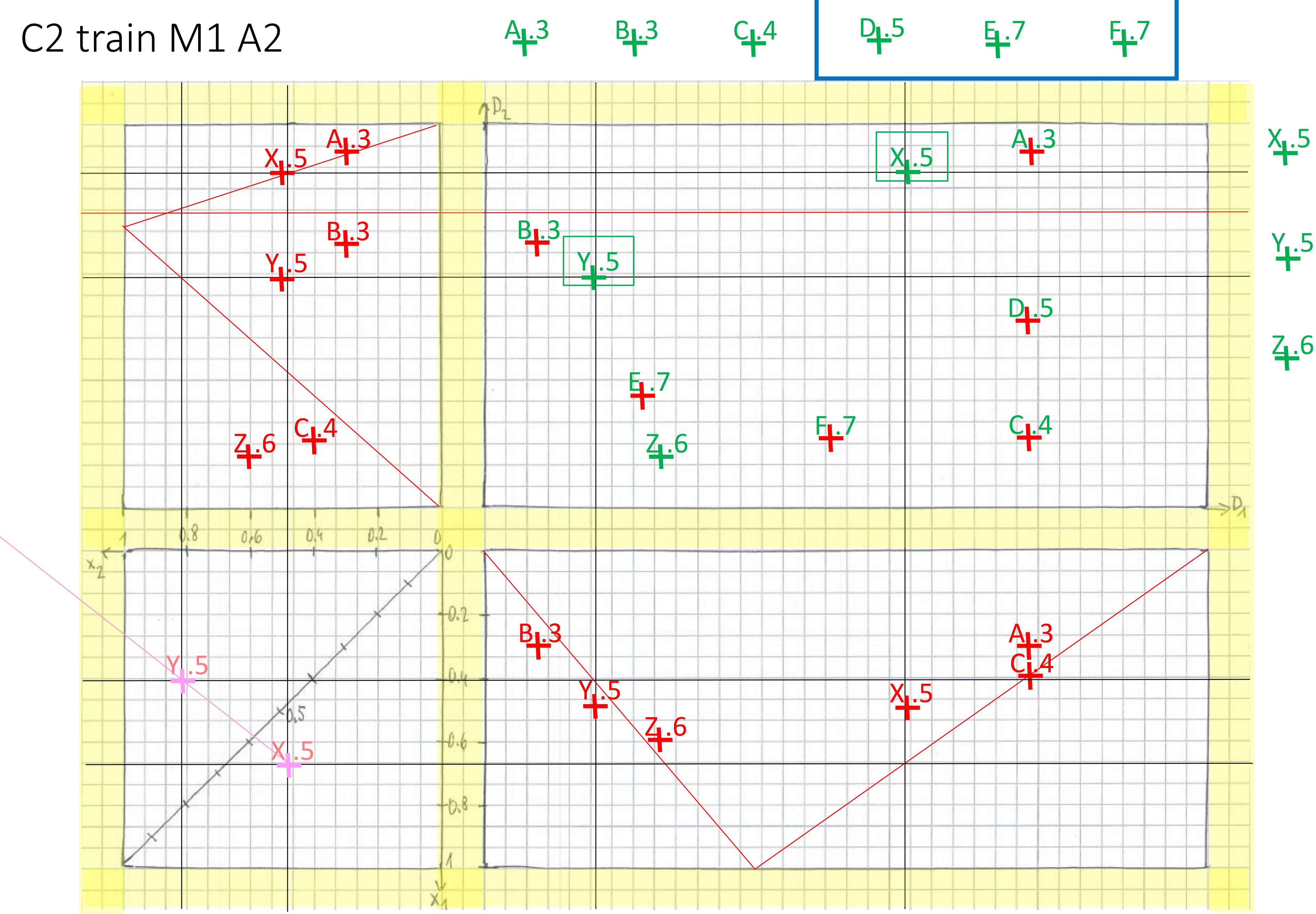
# C2 train M1 A1



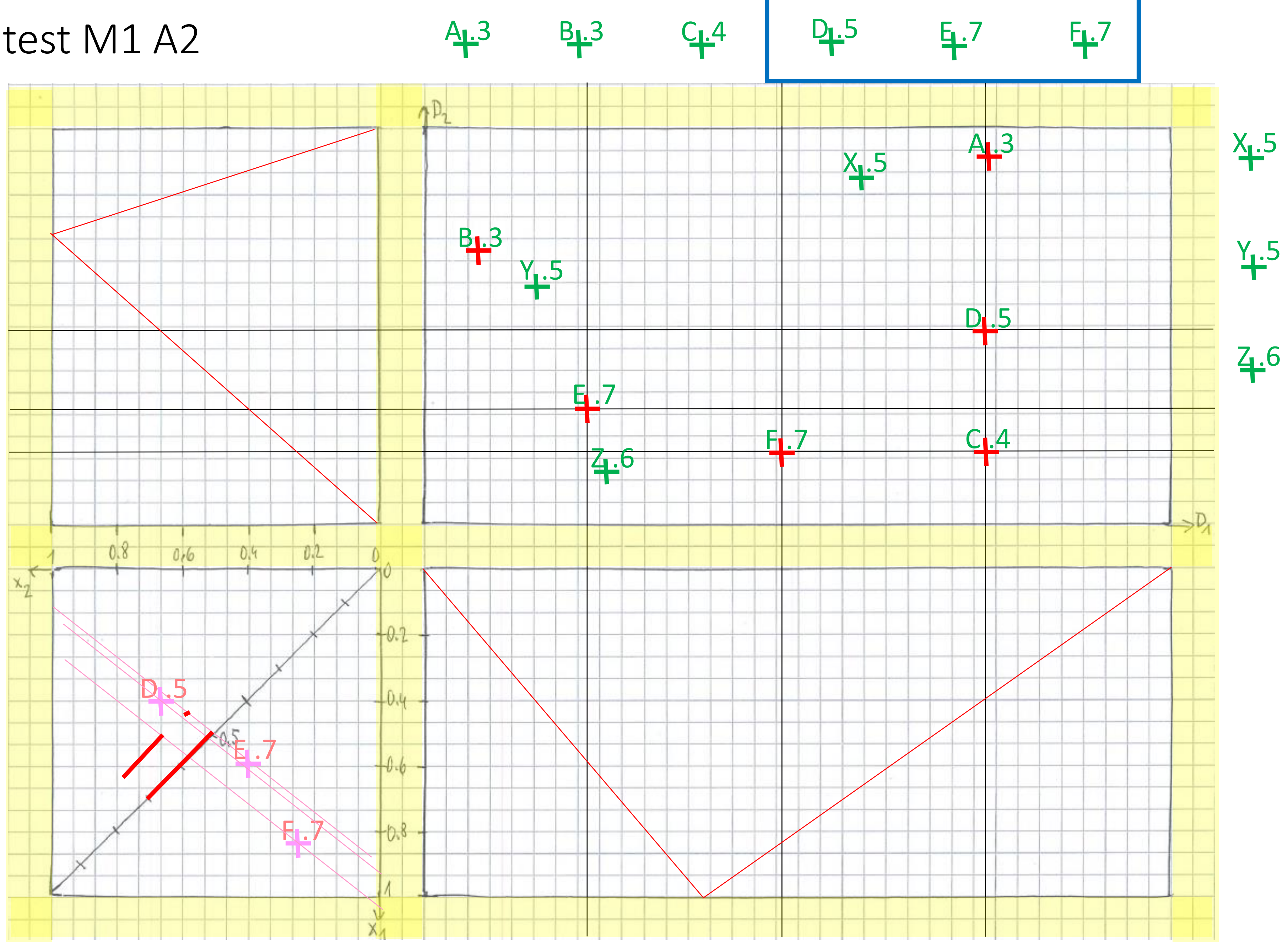
# C2 test M1 A1



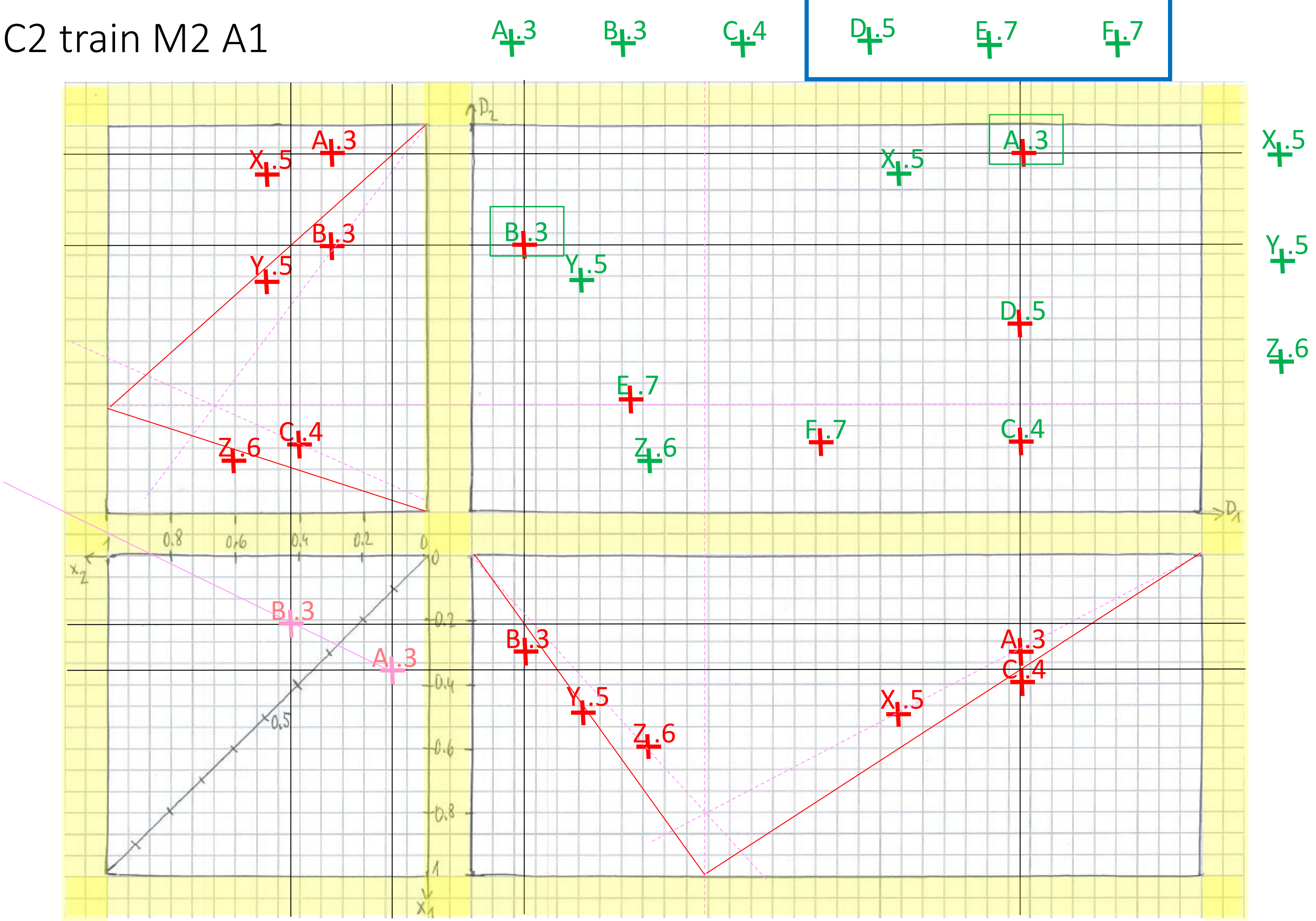
# C2 train M1 A2



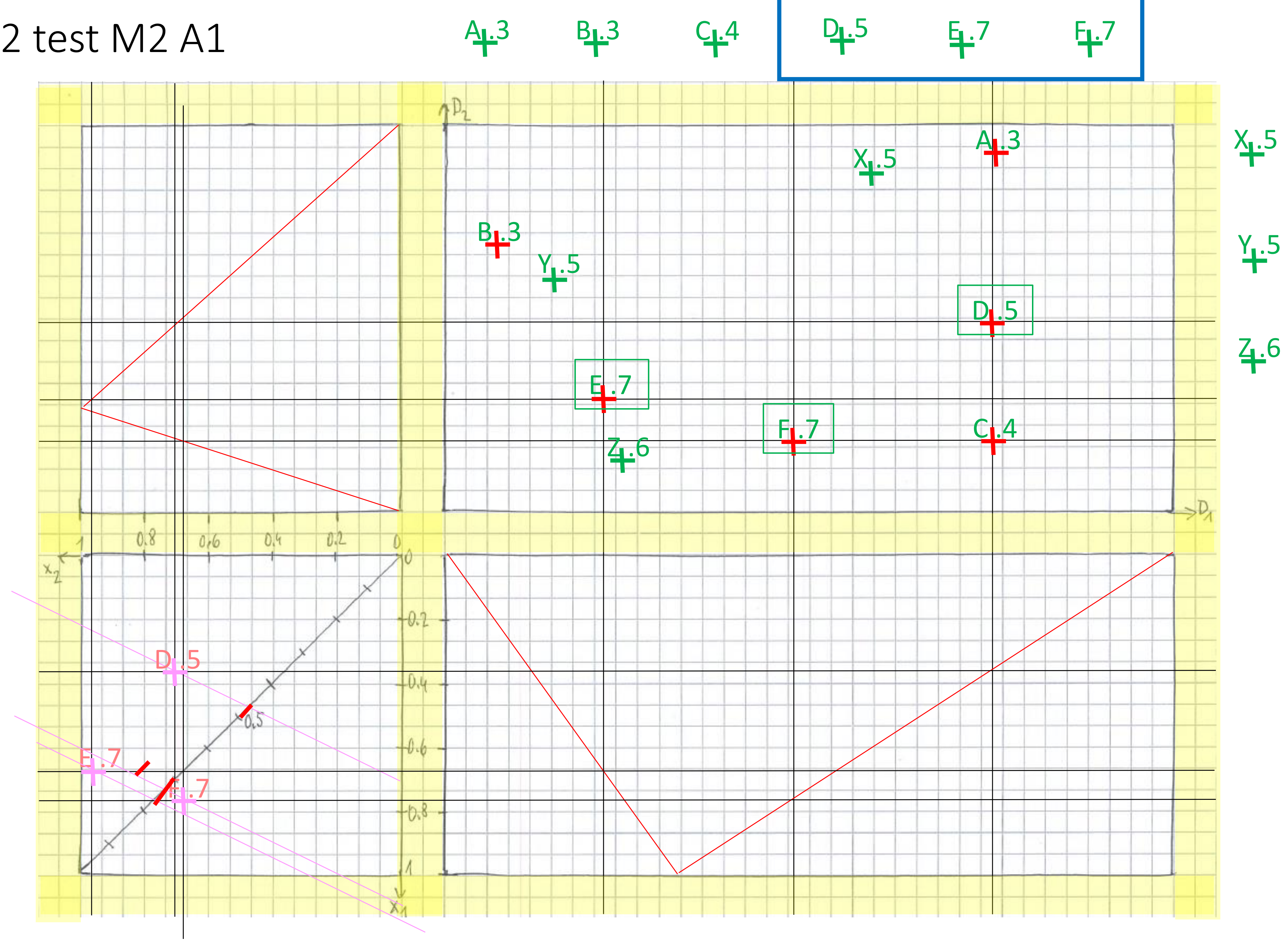
# C2 test M1 A2



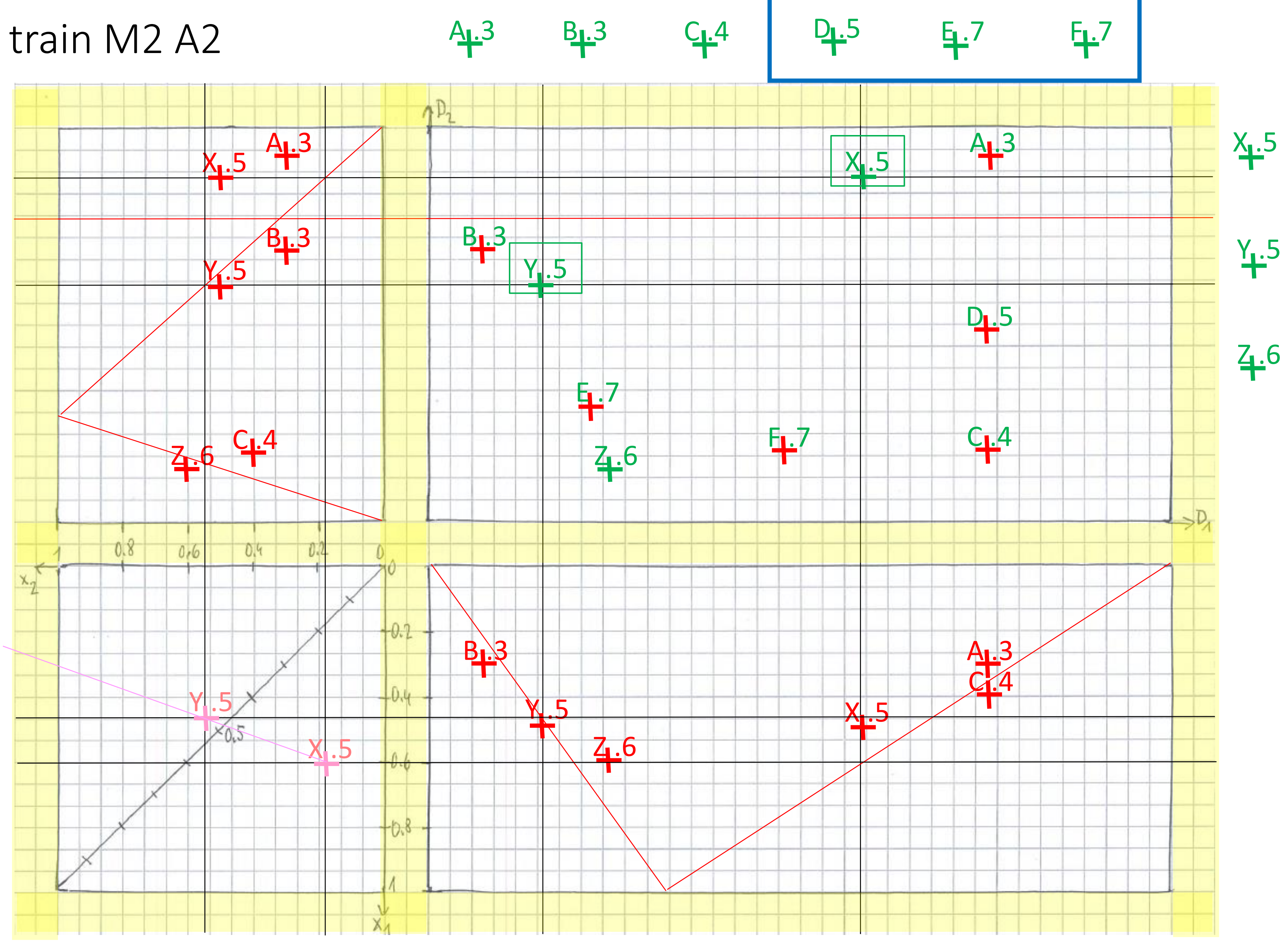
# C2 train M2 A1



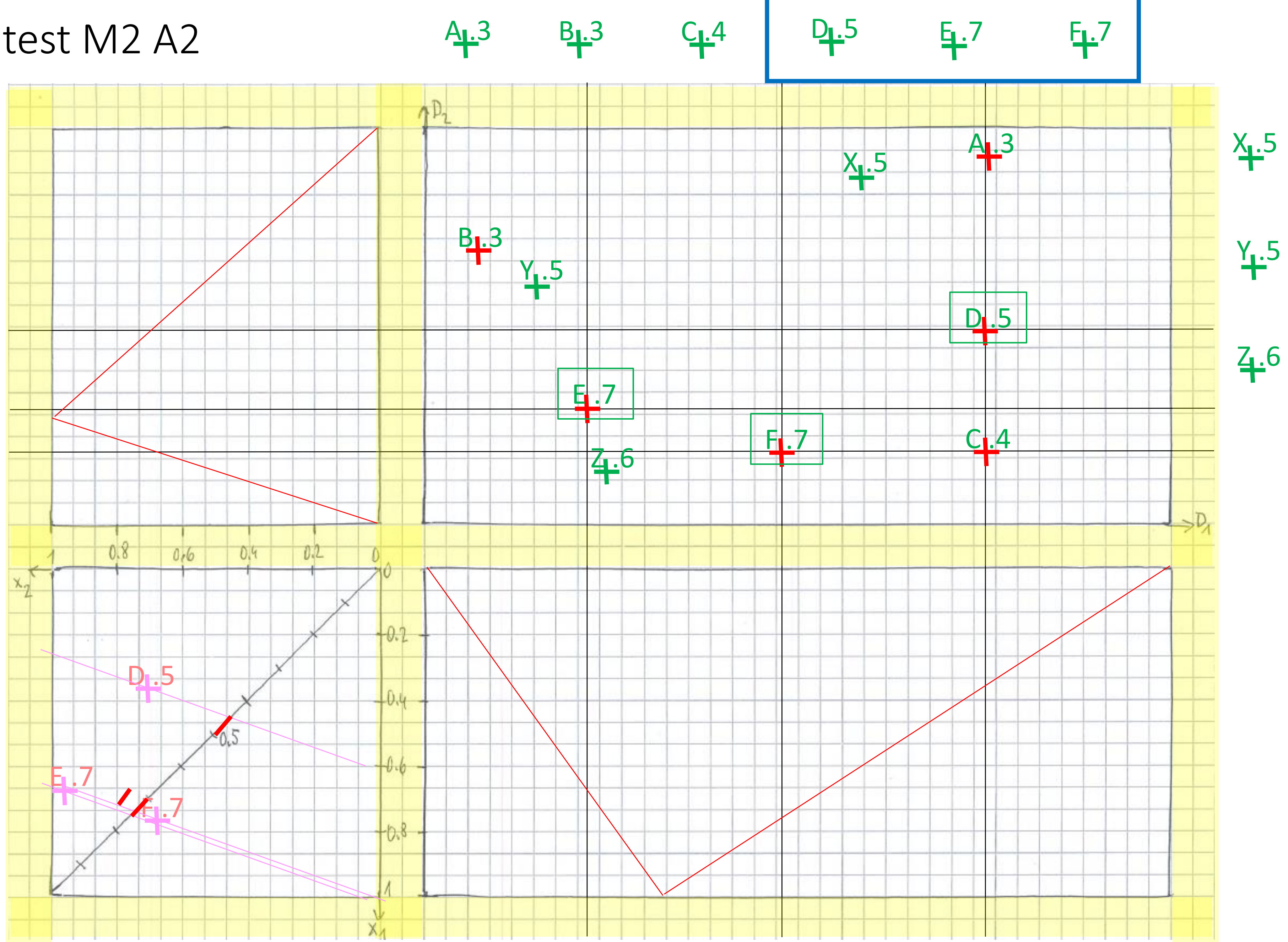
# C2 test M2 A1



# C2 train M2 A2



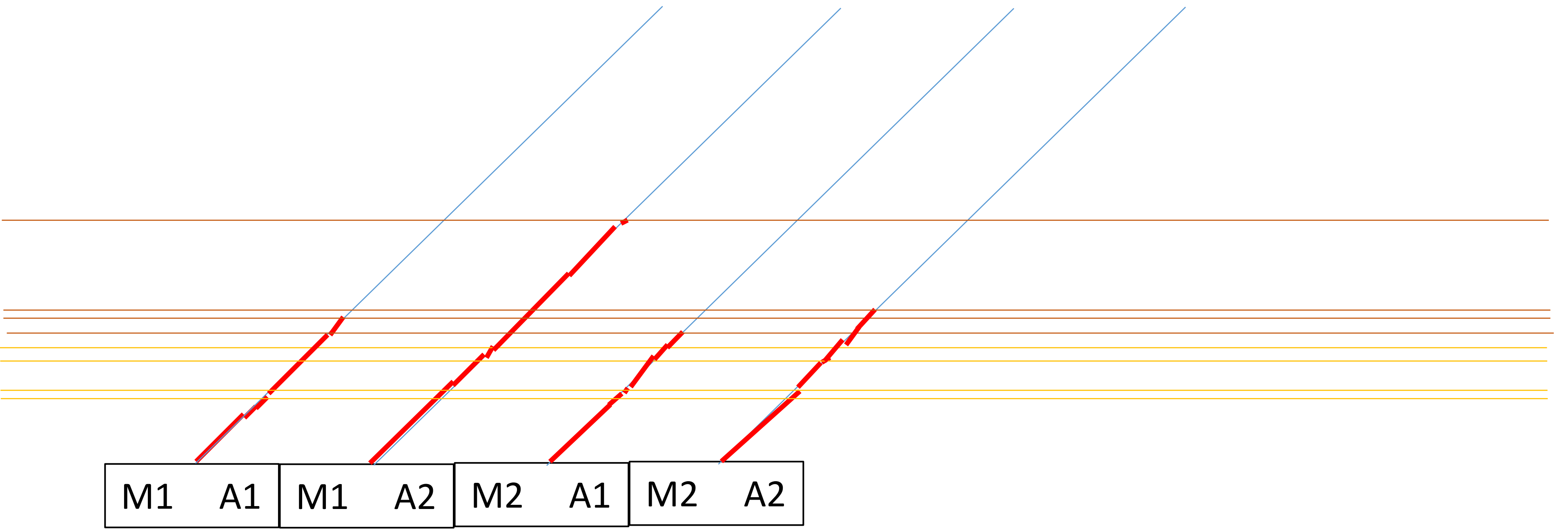
# C2 test M2 A2



A .3
B .3
C .4
D .5
E .7
F .7
X .5
Y .5
Z .6
$D_2$
$D_1$
0,8
0,6
0,4
0,2
0
1
$x_2$
$x_1$
0,5
D .5
E .7
F .7
0,2
0,4
0,6
0,8

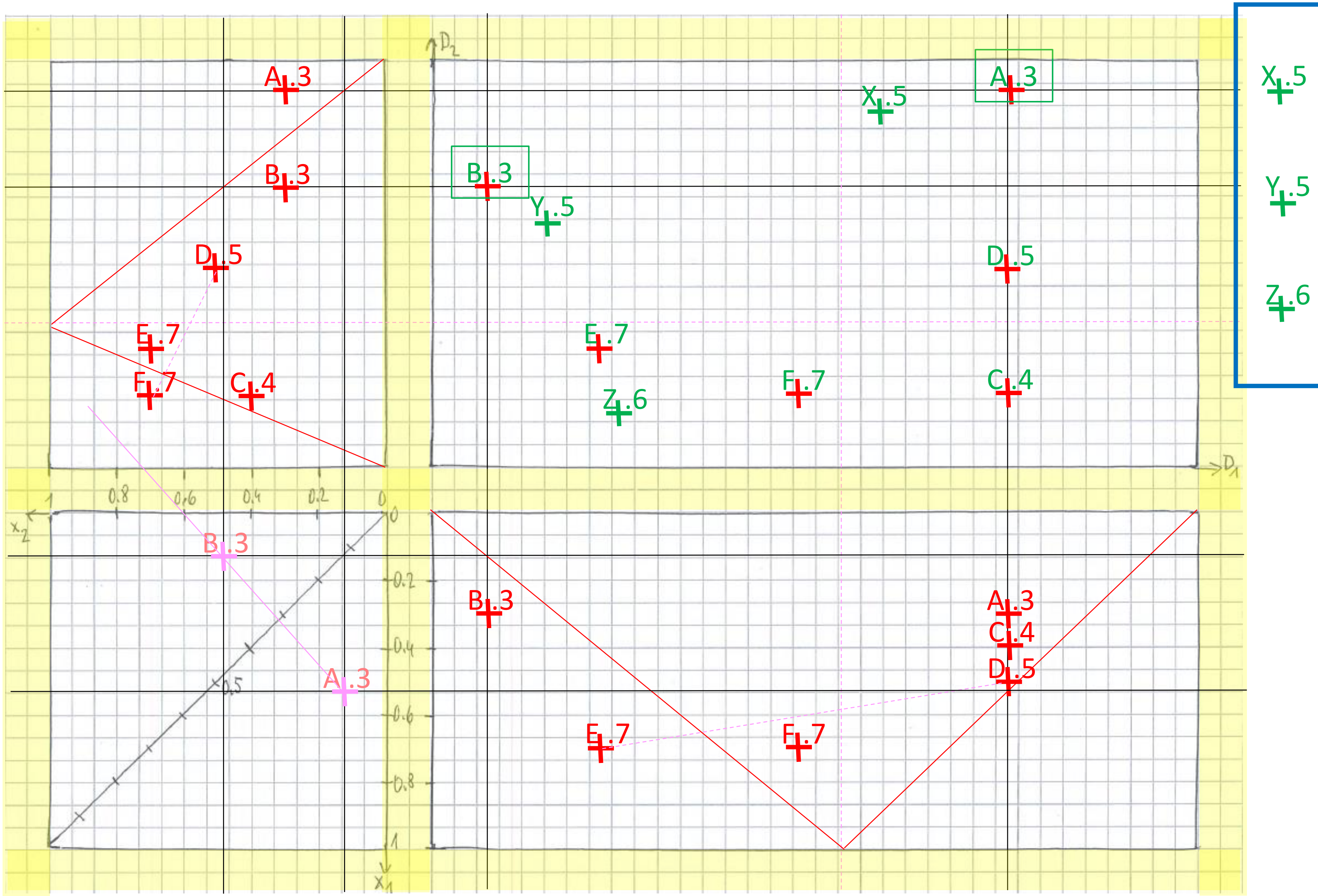
# Experiment evaluation

sum of error measures of splits c1, c2,

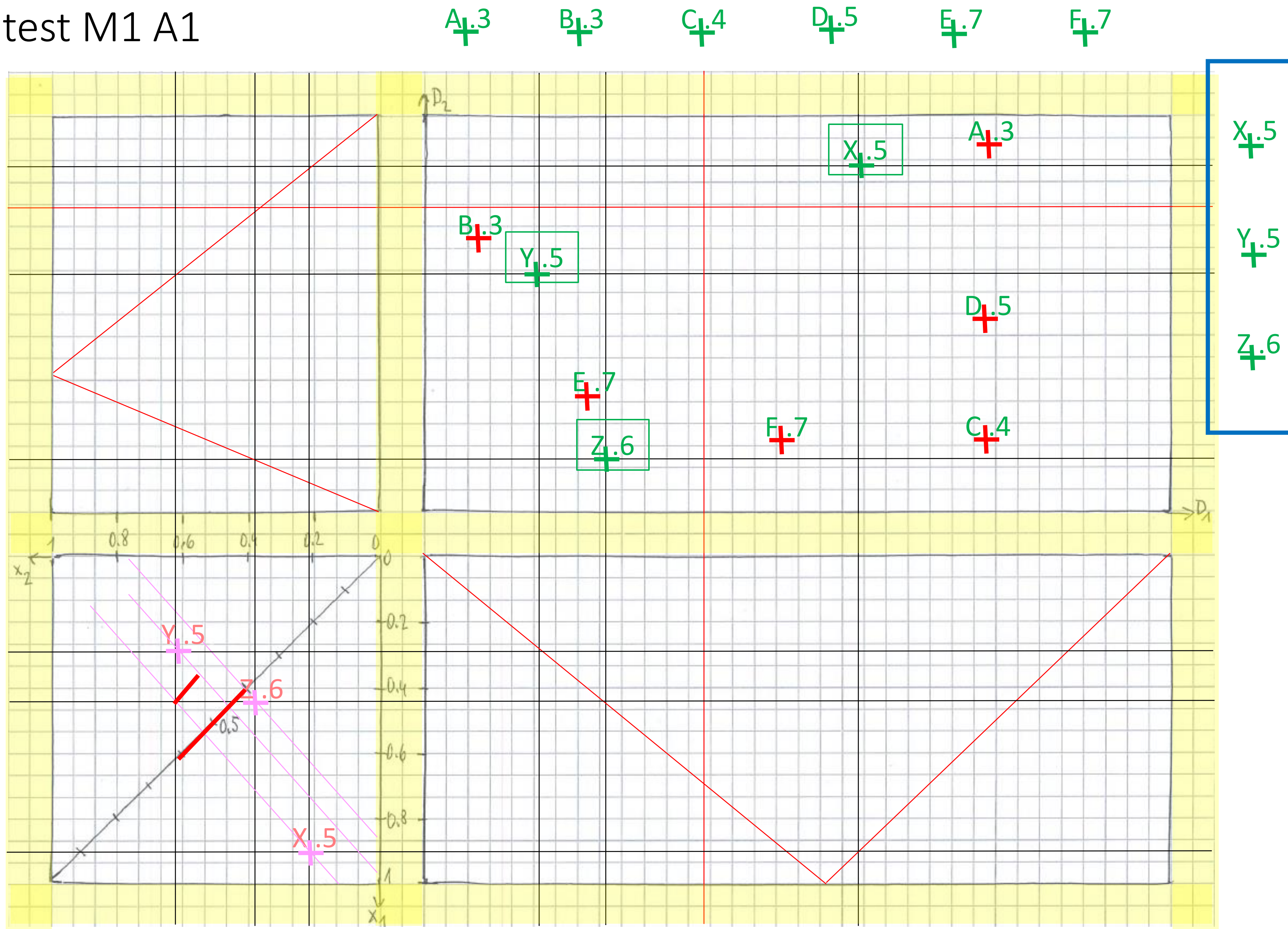


# C3 split

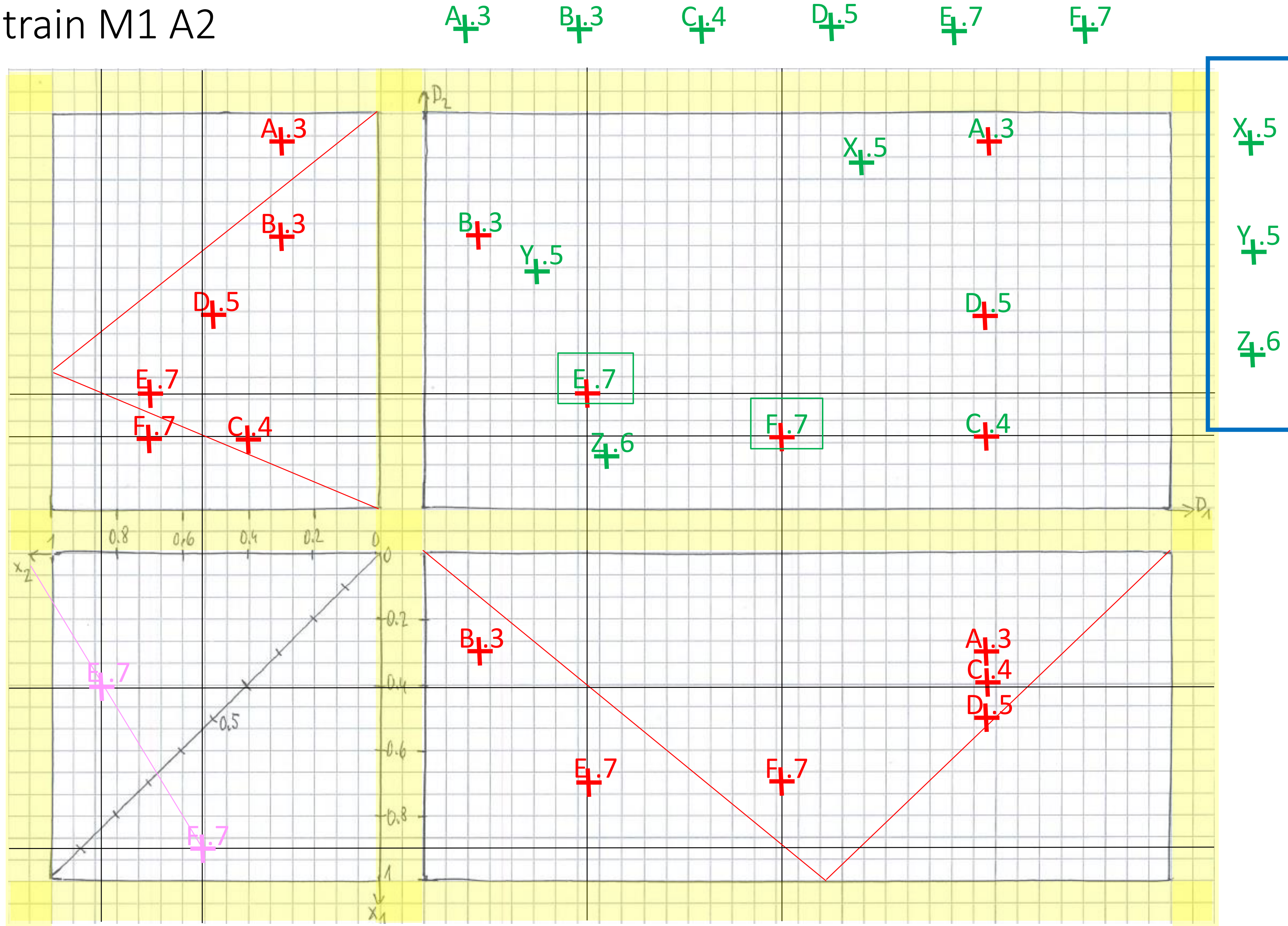
# C3 train M1 A1



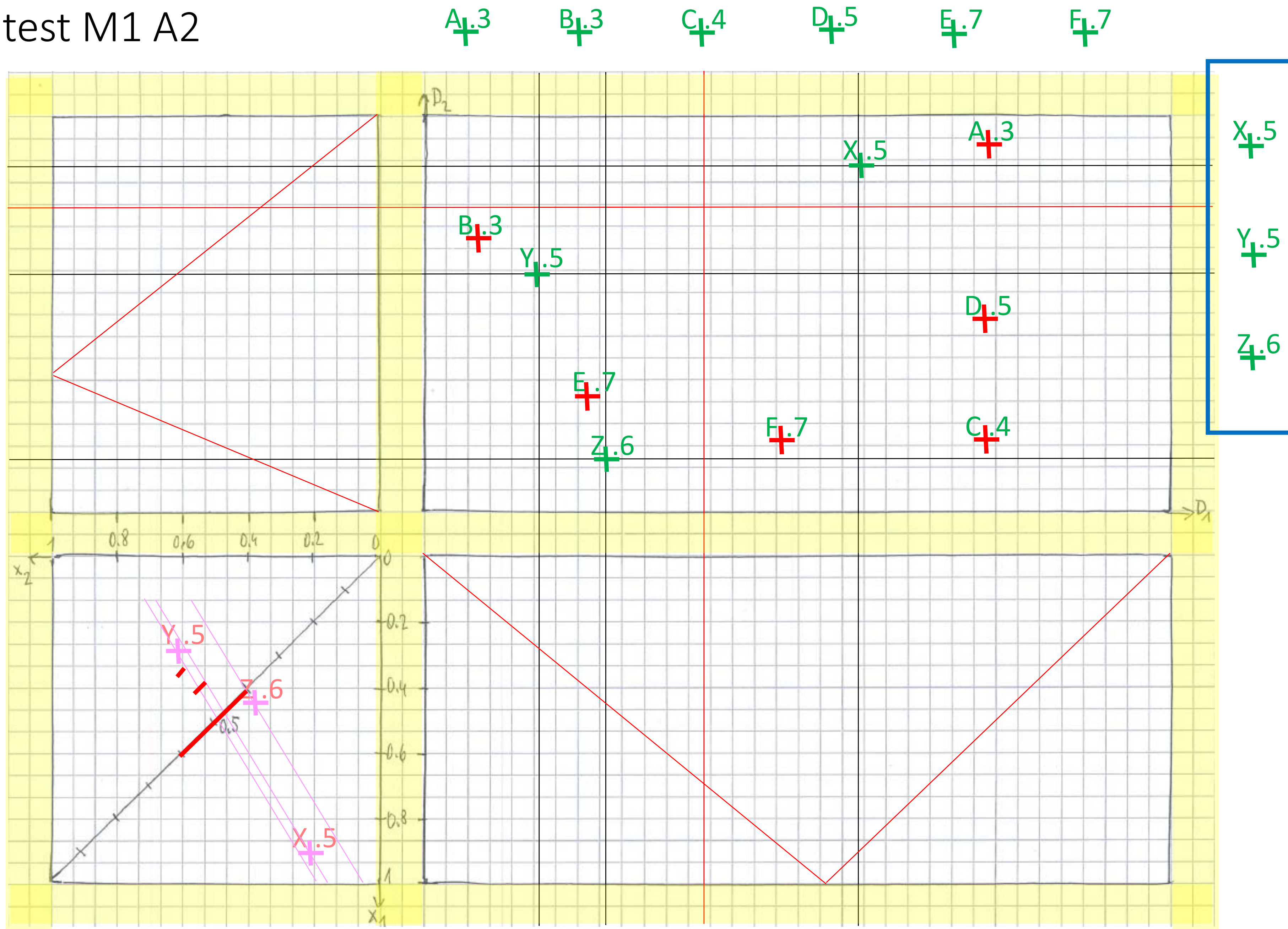
# C3 test M1 A1



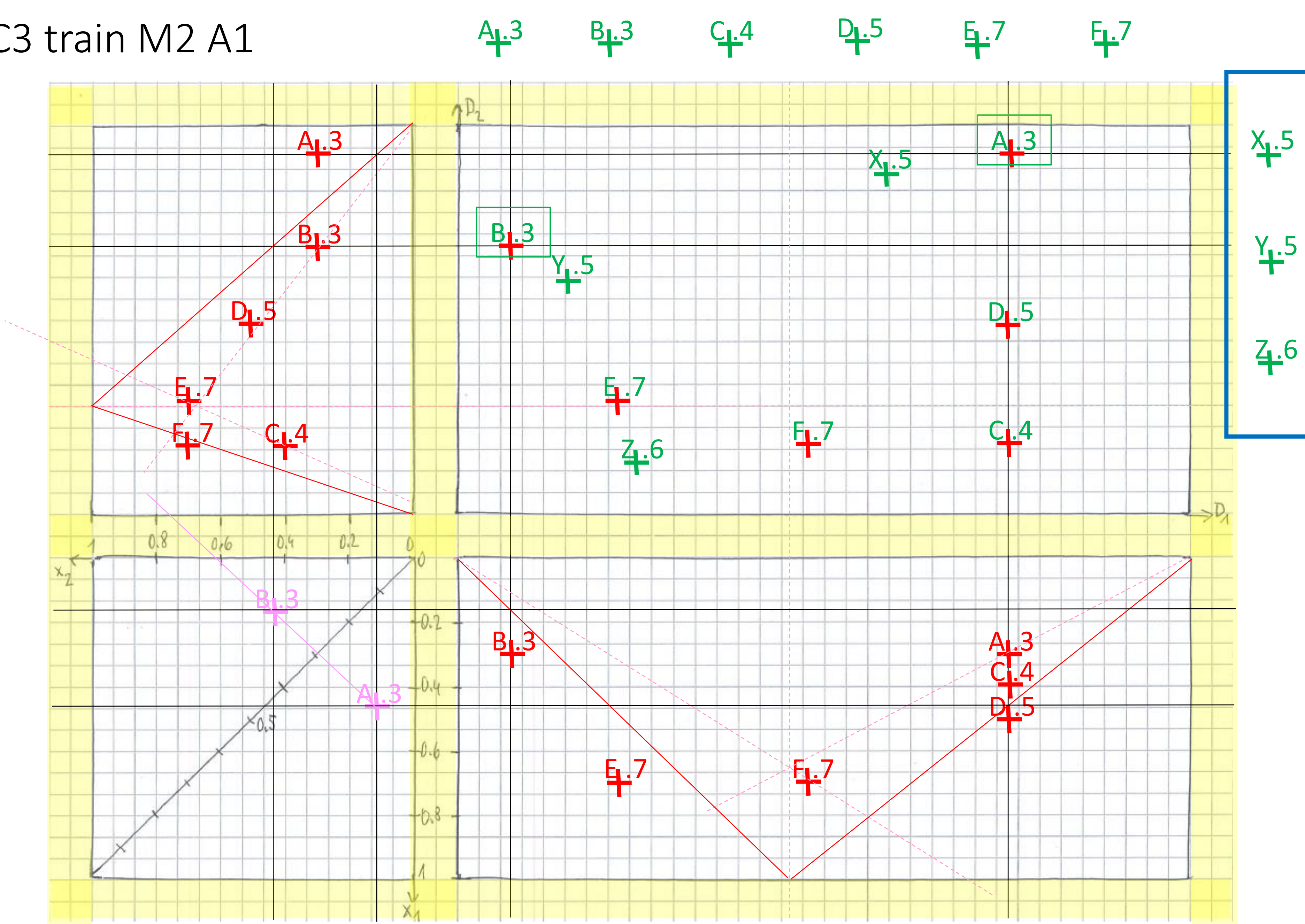
# C3 train M1 A2



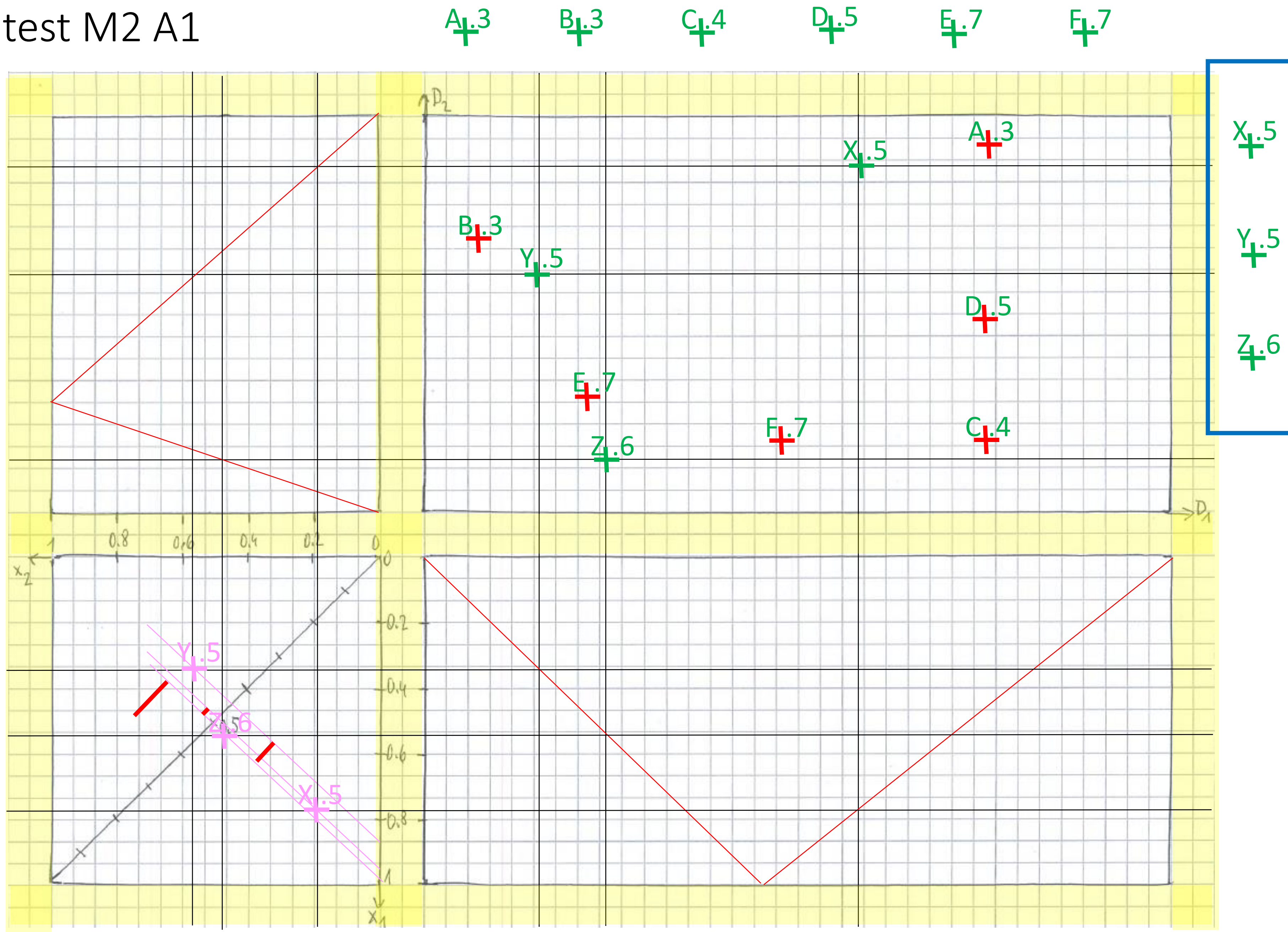
# C3 test M1 A2


A.3
B.3
C.4
D.5
E.7
F.7
X.5
Y.5
Z.6
D2
D1
x2
x1
1
0.8
0.6
0.4
0.2
0
0.2
0.4
0.6
0.8
1
0.5
A.3
X.5
B.3
Y.5
D.5
E.7
F.7
C.4
Z.6
Y.5
Z.6
X.5

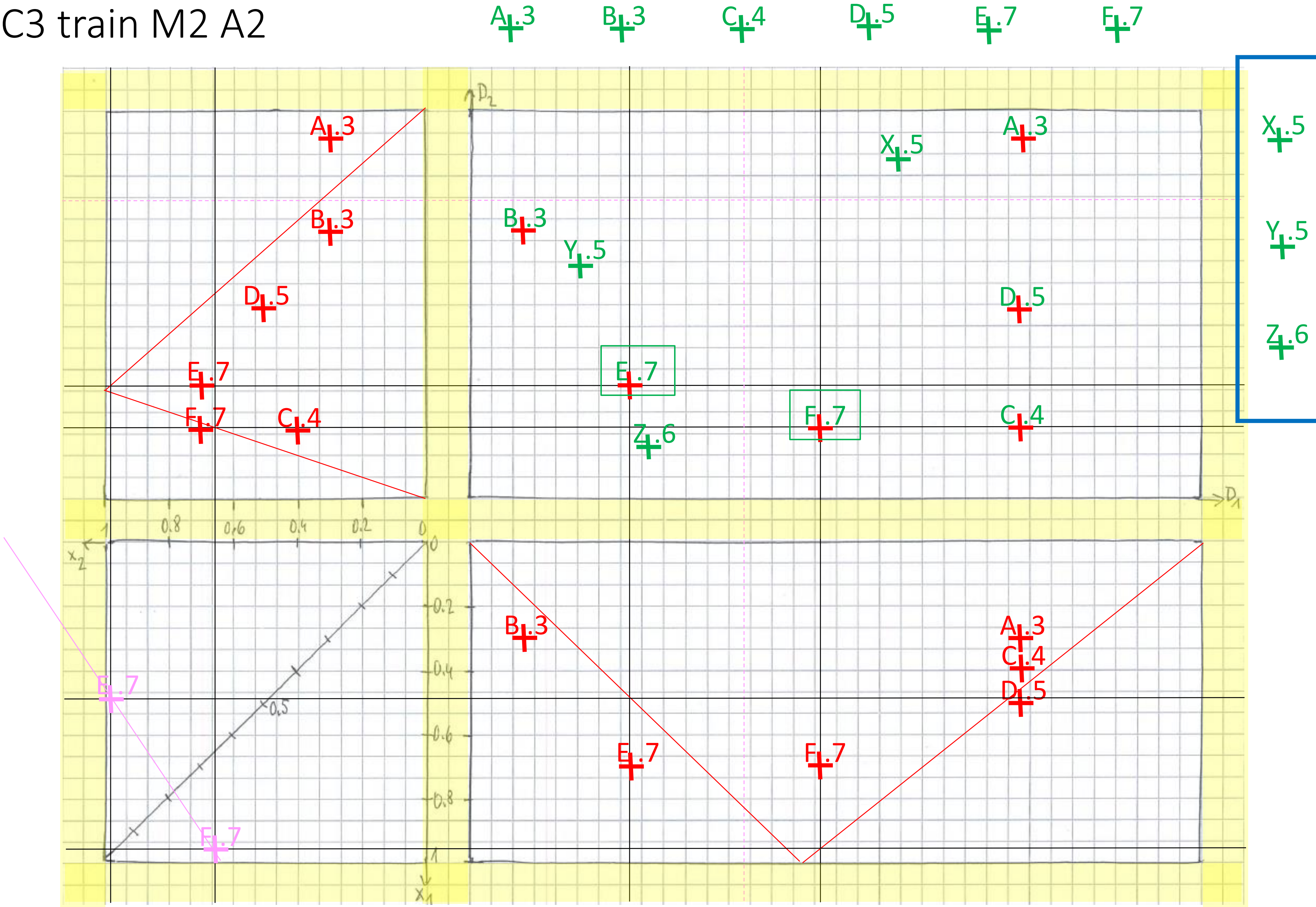
# C3 train M2 A1



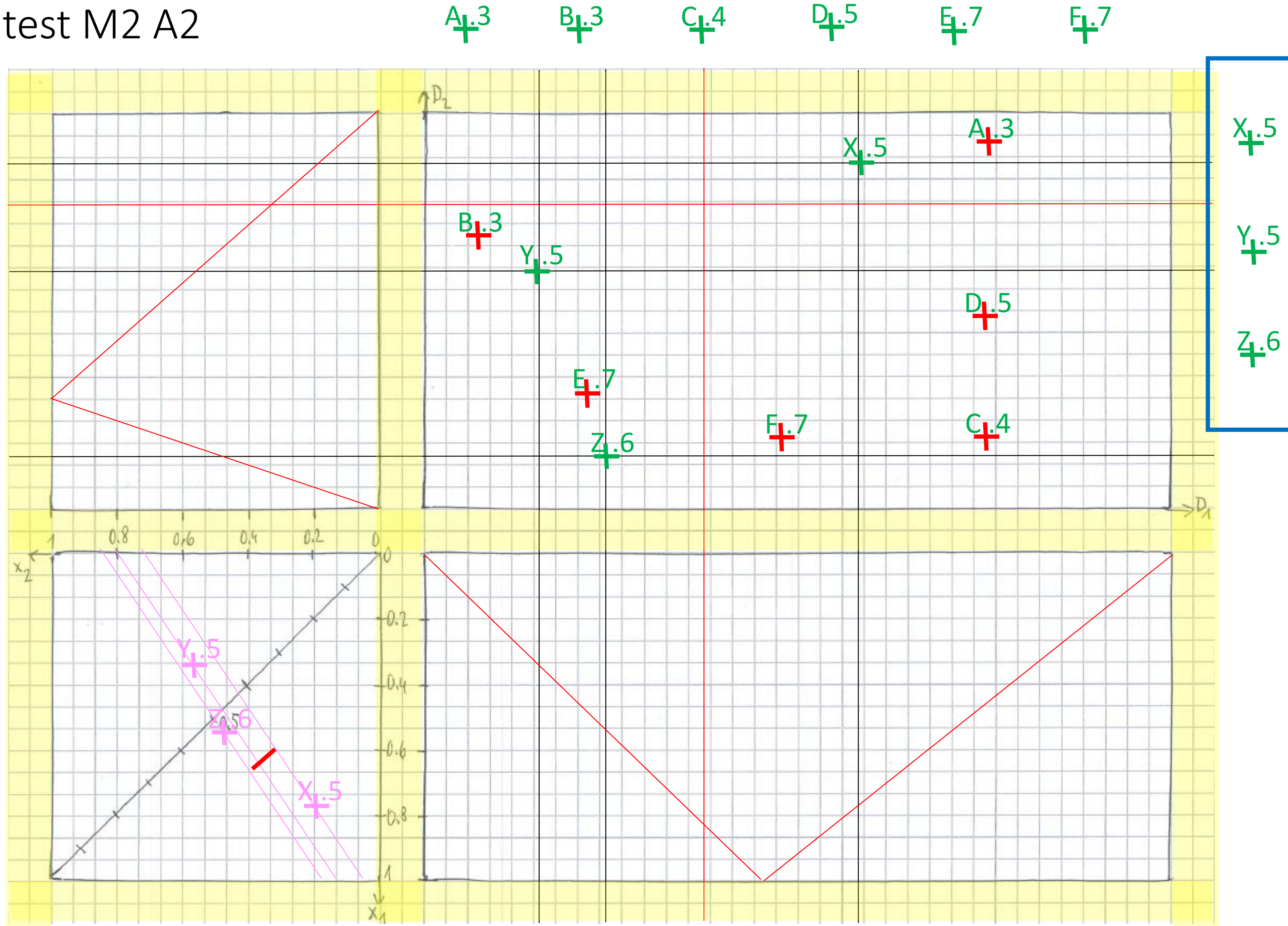
# C3 test M2 A1



# C3 train M2 A2



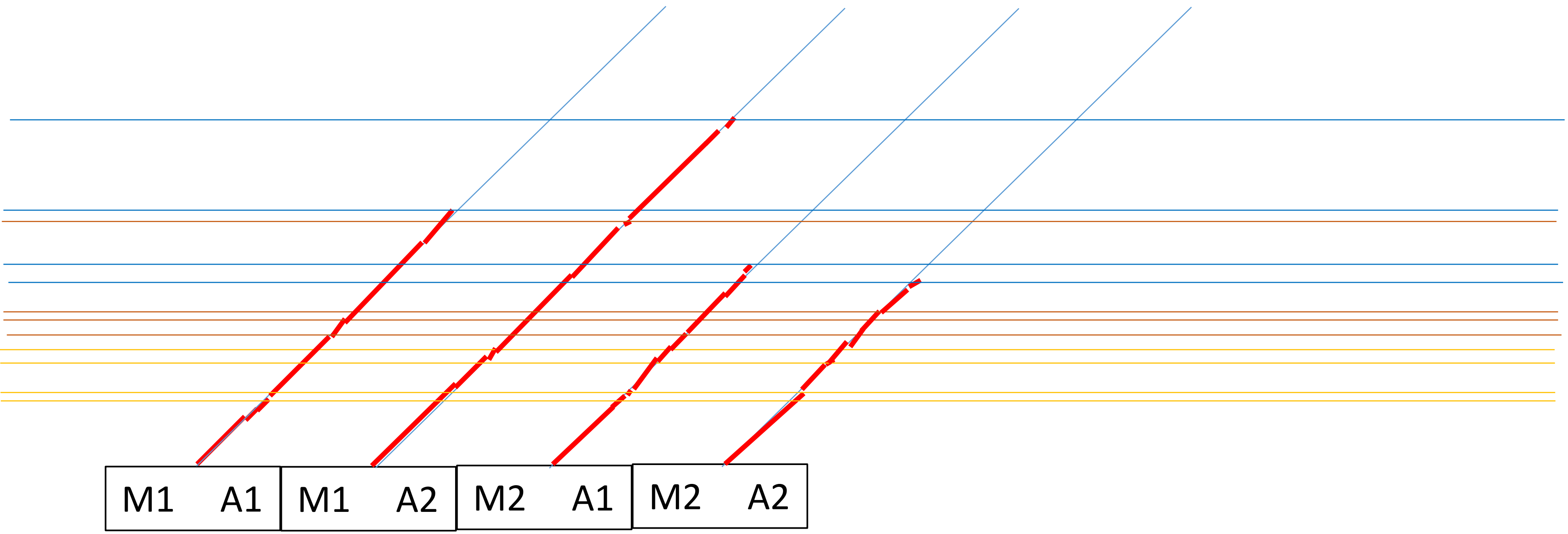
# C3 test M2 A2


A.3
B.3
C.4
D.5
E.7
F.7
D2
A.3
X.5
B.3
Y.5
D.5
E.7
F.7
C.4
Z.6
D1
0.8
0.6
0.4
0.2
0
1
x2
0
0.2
0.4
0.6
0.8
1
x1
Y.5
Z.6
X.5
X.5
Y.5
Z.6

# Experiment evaluation

sum of error measures of splits c1, c2, c3

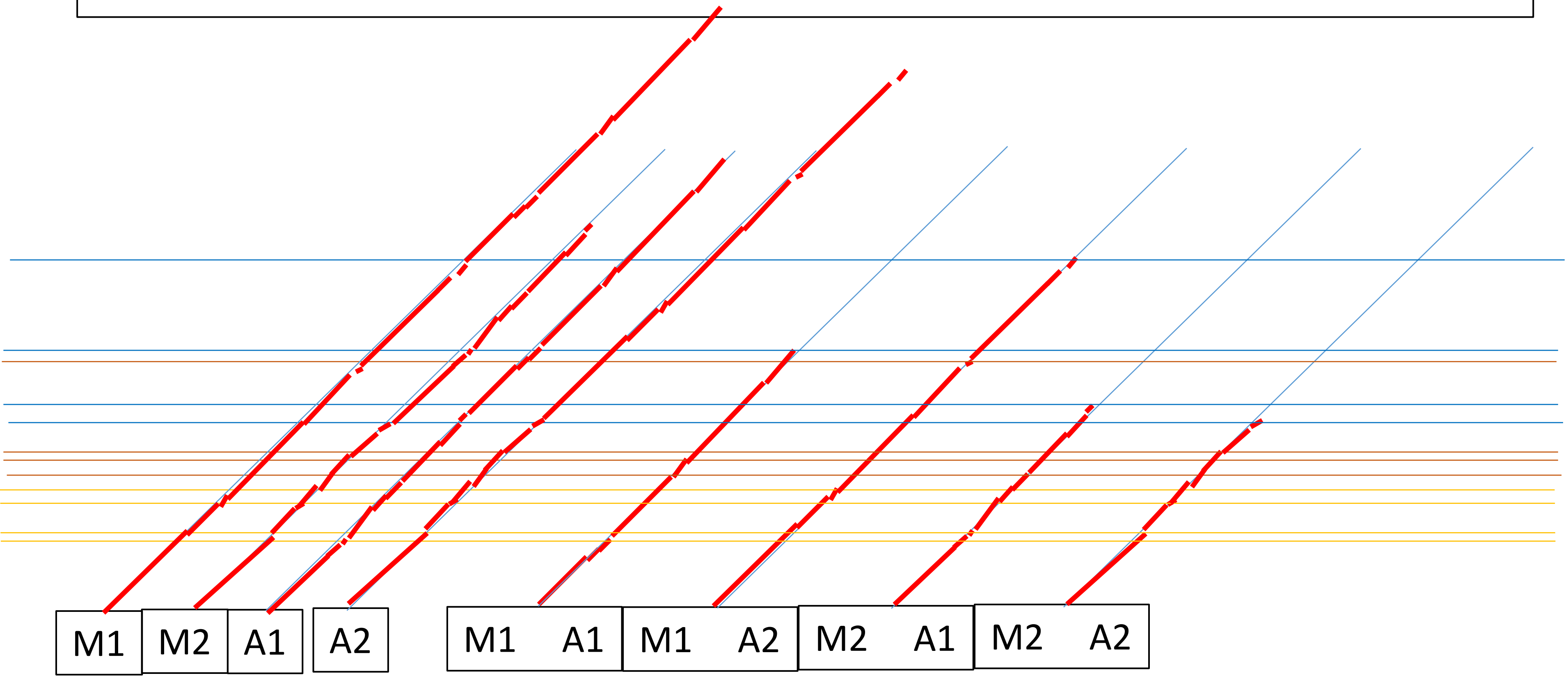
M2+A2 is the winner (tightly followed by M2A1, then M1A1 and far away last M1A2)



# Experiment evaluation

sum of error measures of splits c1, c2, c3

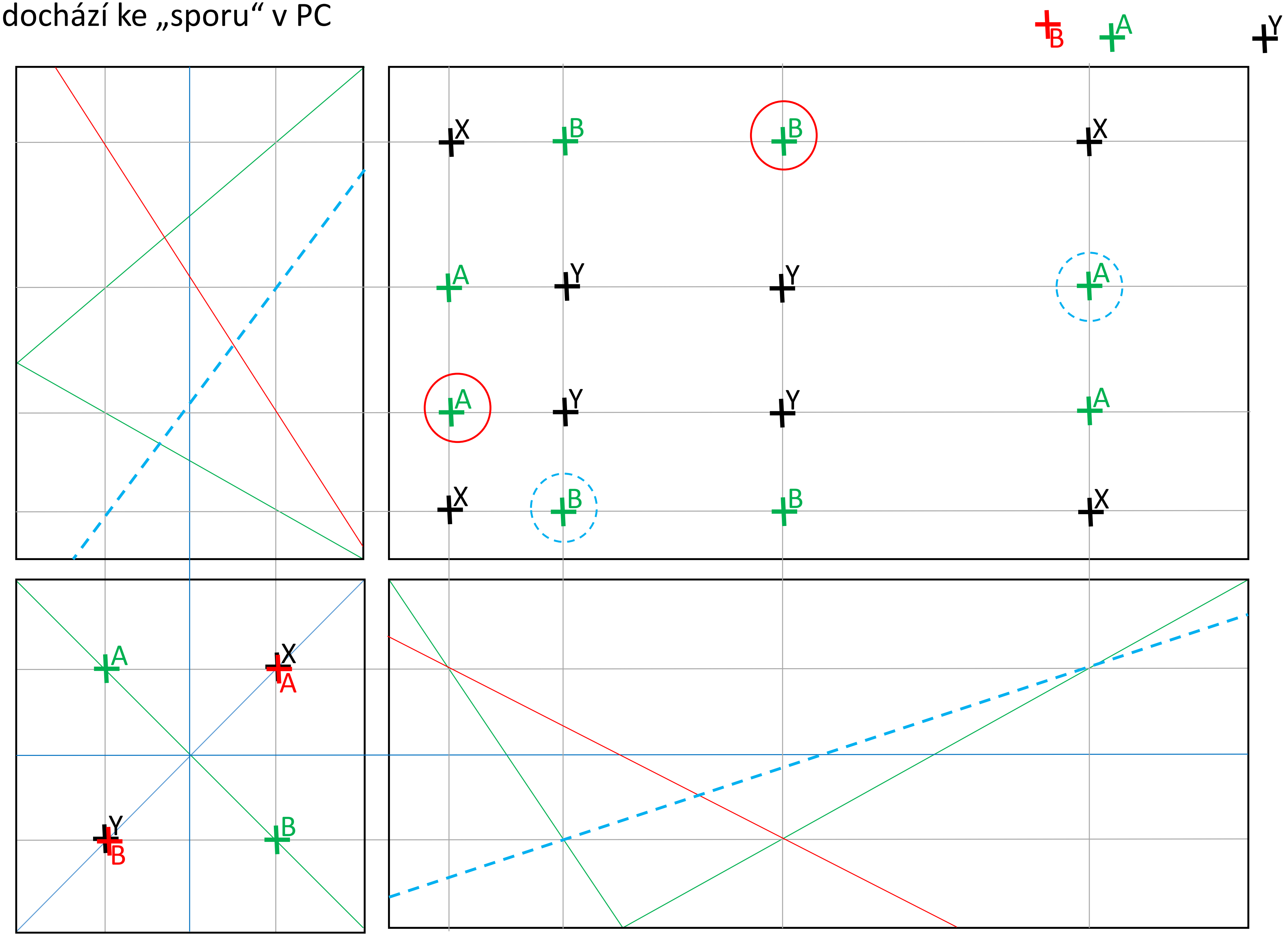
M2+A2 is the winner (tightly followed by M2A1, then M1A1 and far away last M1A2)
In general A1 is little bit better than A2, and M2 is better than M1



# Lessons learned 1?

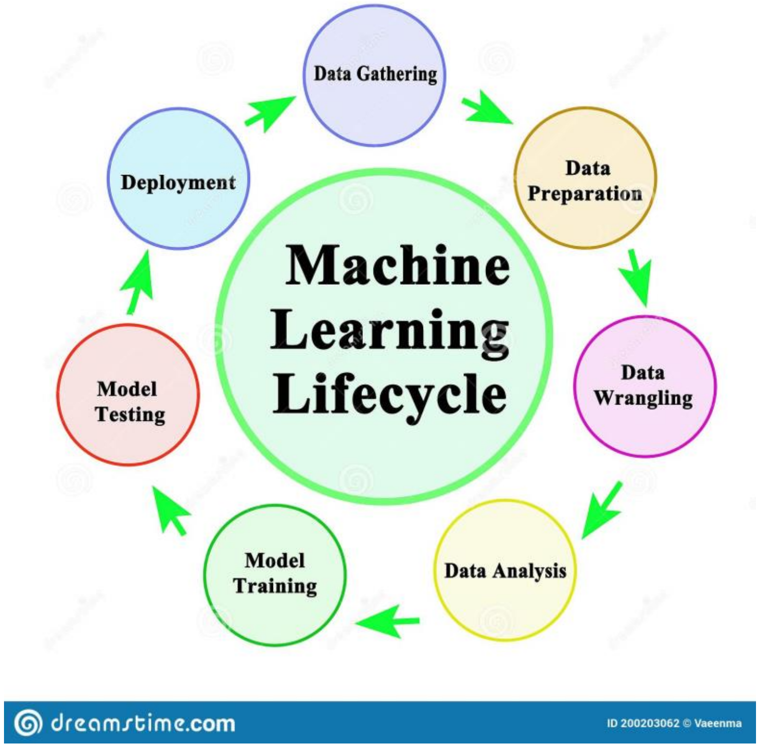
- Are these synthetic data representative?
  - Experience/practice on different data distribution shows that estimated aggregation need not have nonnegative weights (contour line direction goes from SW to NE)
  - We did not specify graphical algorithm for valley/hill
- We do not use information from DC, only from 1D projections – we lose information on mutual interplay of large preference in one attribute with other – there is a need for further algorithms
- Error at highly preferred objects (recommended) is more important than that of low preferred
  - This can be measured by some classification metric
- When designing a graphical algorithm, we (**as humans** with global parallel visual perception) **are tempted** to “guess” the right solution – please erase “green” info

# Jak dochází ke „sporu“ v PC



# Machine learning – data mining - …

- Teacher – labeling
  - Supervised
  - Unsupervised
- How much do I know about items
  - Content based
  - Collaborative
  - Hybrid
- How much do I know about user
  - Demographic data
  - Historical data
  - Anonymous users




- Domain specific (leisure, frequency, … )
- For more see lectures of Ladislav Peska

# Evaluation metrics influences all

- **Regression** tasks
  - On test data compare $r^{f,t}$ with $r^u$
  - Distance of $r^u$ and $r^u{}_e$ as for f, g : O → [0, 1]

$$RMSE(f,g) = \sqrt{\frac{\sum (f(o)-g(o))^2}{\#O}}$$

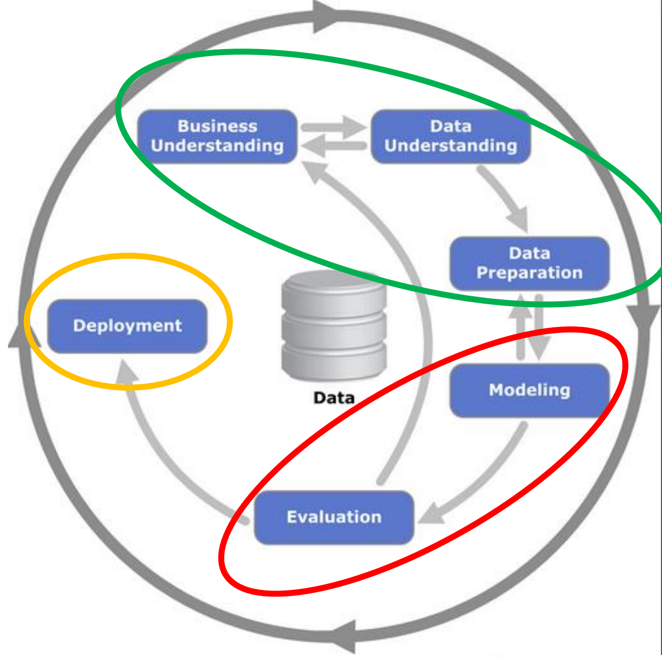
$$L_2(f,g) = \sqrt{\sum (f(o)-g(o))^2}$$

$$L_1(f,g) = \sum |f(o)-g(o)|$$

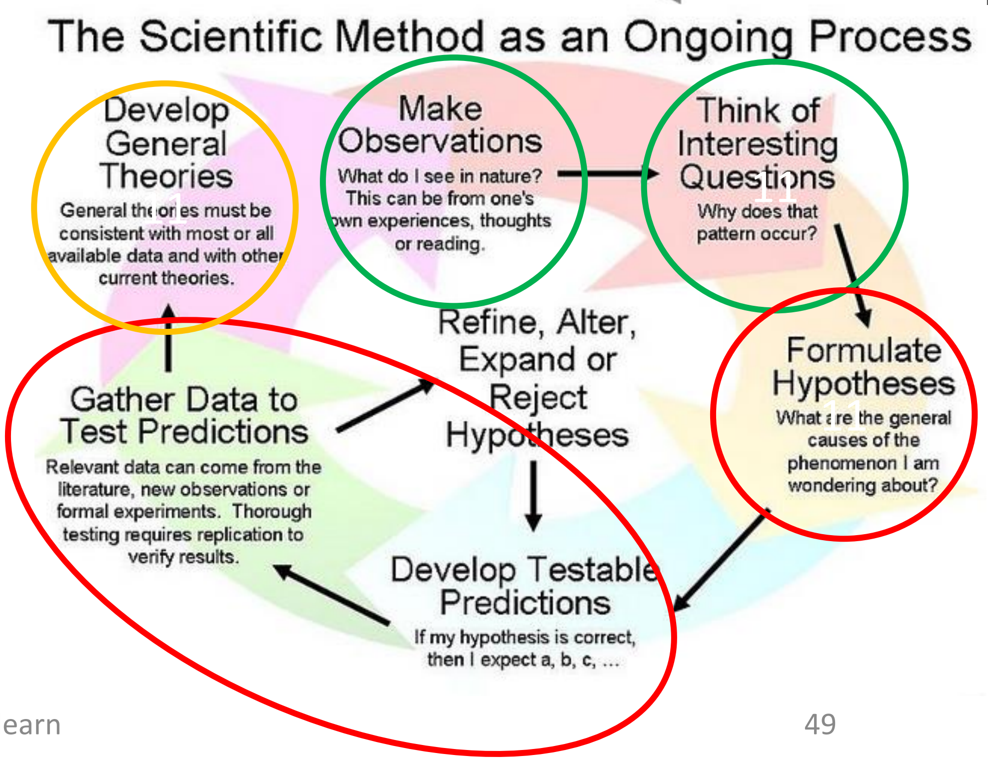
$$sim(f,g) = AVG(|f(o)-g(o)|)$$

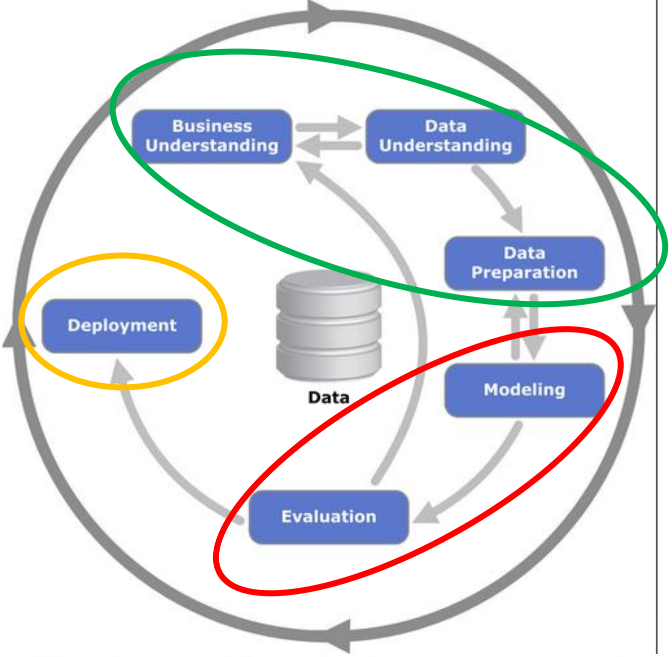
- **Classification** tasks
  - SARS-CoV-19 positive/negative – reality/test

# Recall - scientific method
# Physical nature is unique
# Users’ nature not unique




- Scientific method
  - Observation/Research
  - Hypothesis
  - Prediction
  - Experimentation
  - Conclusion
- Current solutions
- Components
- Context
- ?forgot something?

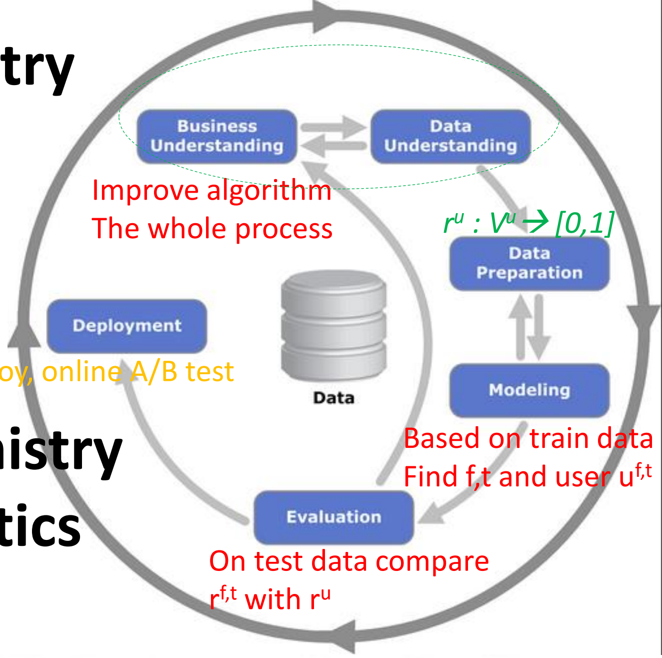


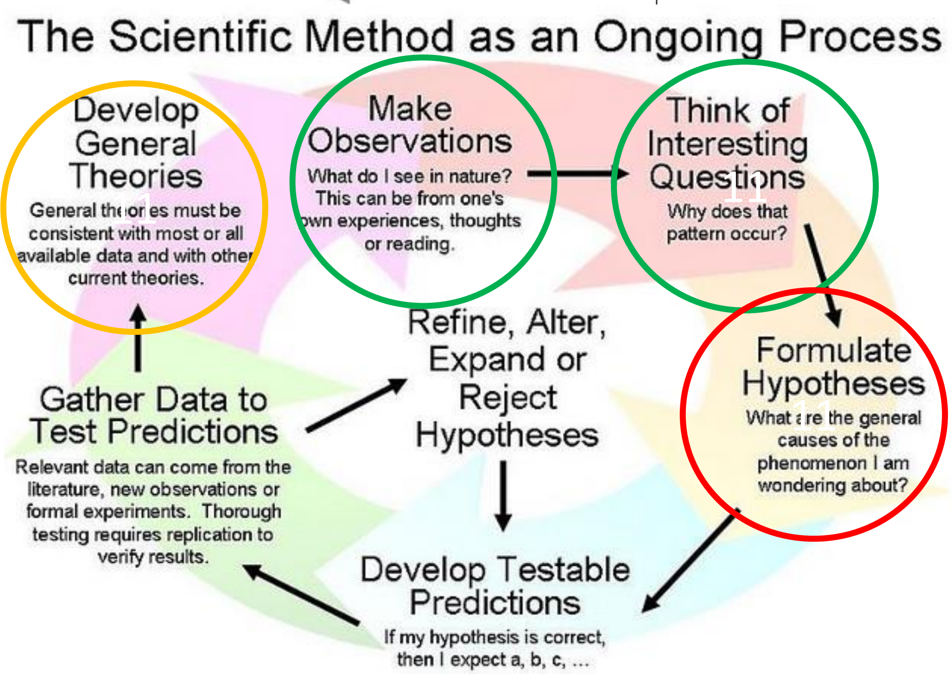


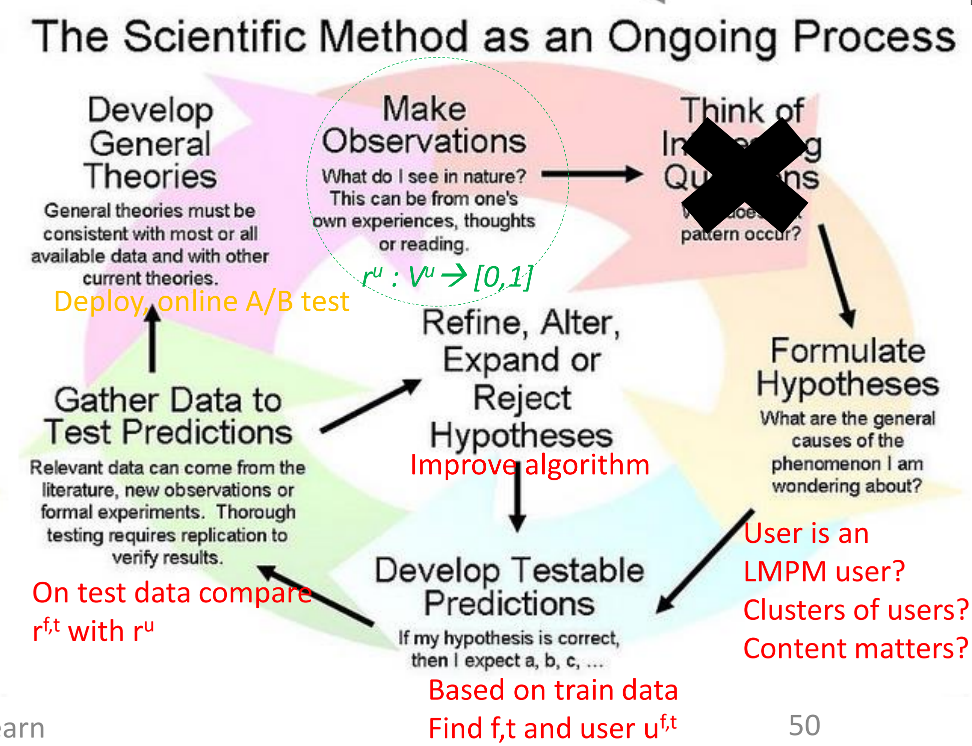

# CRISP-DM Cross Industry Standard Process for Data Mining

# Scientific method

**physics, biology, chemistry**
**medicine, pharmaceutics**
**Sociology, psychology**



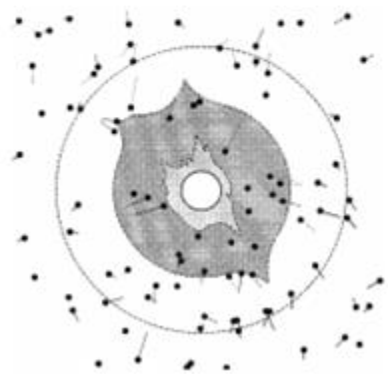
# Metric – regression, order, business

- Classical
  - RMSE, ABS, AVG, $L^p$ , …
  - Pearson , …
  - … on all objects, top-k,
- Order matters
  - nDCG, Kendall, Average position of best object,
  - 1-hit, Next -1, 1st hit, …
  - Again parametrize wrt k in top-k
- What are business relevant metric? dynamic, session, ..
  - E.g. for Netflix it is loyalty ((no)content, explicit rating, user identified by registration)
  - For recommendation based on implicit user behavior, no registration, ...
  - Epidemic / pandemic (test PCR/Ag, has virus, is contagious, has symptoms, needs hospital treatment, serious, ... (dead))

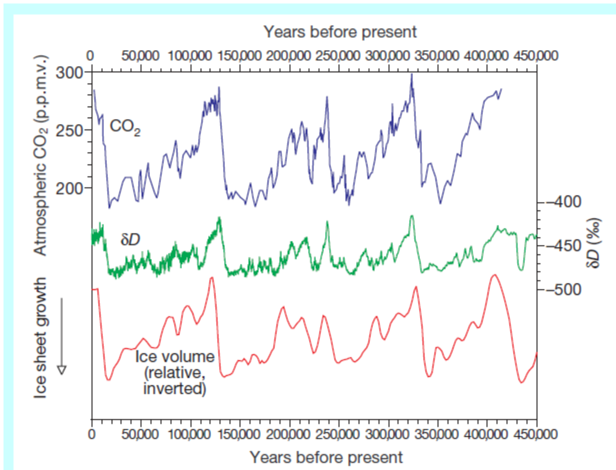
# Deduction – induction – abduction – analogy …

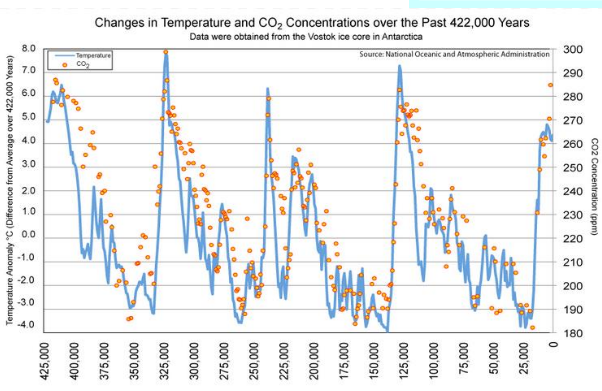
- Measures of success, mathematics, statistics, computer science, industry, customers, economy …
- Hypothesis should be refutable
- The International Congress of Logic, Methodology and Philosophy of Science and Technology (*CLMPST*)
- Strategic planning
- Where is the value
  - Innovation, patent, know-how ownership
  - Investment
  - Assembly – Montagewerk – montovna
  - Final product – added value

# Measure, compute(simulate), media, politics



Change in star position during the eclipse




Eddington confirms Einstein's theory of **relativity** - data - 1919 eclipse

Effect of human activity on **climate change** insignificant

The most viable hypotheses for the cause of glacial/interglacial **CO2** change involve ... **by biological production**

Peter Coles:... But the media **don't seem to like representing science** the way it actually is, ... They prefer instead to portray scientists as priests, laying down the law ...