جلسه هفتم (آزمایشات سه فاکتوره)

آزمایشاتی که اثر سه فاکتور تیماری که هر فاکتور دارای سطوح متفاوت مربوط به خود هست را بررسی می کند، آزمایشات سه فاکتوره گویند. این آزمایشات هم مانند سایر آزمایشات دارای یک طرح پایه برای قالب خود می باشند. این آزمایشات از ساده به پیچیده تر عبارتند از: فاکتوریل سه عامله، فاکتوریل-اسپیلیت پلات، اسپیلیت-فاکتوریل، اسپیلیت-اسپیلیت پلات و اسپیلیت-لسپیلیت پلات در زمان.

تفسیر اثر متقابل:

اثر متقابل برآوردی جدای از اثرات ساده است. بدین مفهوم که وقتی در حال صحبت از یک اثر ساده (مثلا فاکتور a) هستیم، فقط اثر انفرادی سطوح آن فاکتور را بر صفت اندازه گیری شده خواهیم داشت. این اثر ساده علاوه بر اثر انفرادی داری یک اثر متقابل هم هست که نشان می دهد این فاکتور اگر در کنار فاکتور دیگری اعمال شود جواب معنی داری بدست می دهد و در این حالت چون فاکتورهای موجود در آزمایش با هم اعمال شده اند تفسیر بر اثر ساده ممکن است برآوردی اشتباه بدست دهد. بنابراین تنها باید روی ترکیبها بحث و بررسی کرد. مثلا دو فاکتور a و b اگر اثر متقابلشان معنی دار باشد یعنی حداقل یکی از سطوح a در کنار یکی از سطوح b ترکیبی بوجود آورده که نسبت به سایر ترکیبات سطوح این دو فاکتور اختلاف معنی داری دارد. برای کشف این ترکیب یا ترکیبات معنی دار نیاز به مقایسه میانگین داریم و تفسیر روی این اثرات برآوردهای بهتری به ما می دهد (بجای تفسیر بر اثرات انفرادی). در حالتی هم داریم که اثر متقابل معنی دار نیست ولی اثرات ساده آن اثر متقابل، معنی دار هستند. در این حالت این نتیجه حاصل می شود که ترکیب این دو فاکتور با هم چیز خاصی را ایجاد نمی کند و فاکتورها روی هم اثر متقابل ندارند. یعنی ترکیب آن ها چیز عجیبی بدست نمی دهد (اثر افزایشی یا کاهشی متقابلی ندارند) و در این حالت برای افزایش محصول باید هر فاکتور را جداگانه بررسی کرد و بهترین سطوح هر فاکتور راانتخاب نمود. در حالی که در حالت قبل بهترین ترکیب را انتخاب می نمودیم. گاهی هم نه اثر متقابل و نه یکی از اثرات ساده معنی دار نیستند و به عنوان مثال فقط اثر ساده a معنی دار است در این حالت ما عامل b را برای مطالعات بعدی حذف می کنیم زیرا فهمیده ایم این عامل نه به تنهایی و نه در ترکیب با عامل a اثری روی صفات ما ندارد.

داده های نمونه:

این قانون ثابت مطرح است که تعداد ستون تعریف برابر است با تعداد فاکتورها به علاوه تعداد نوع تکرار بندی در طرح پایه. مثلا در یک آزمایش سه فاکتوره در قالب بلوک یا کاملا تصادفی. ما چهار ستون تعریف داریم سه تا به ازای سه فاکتور و یکی به ازای تکرار. اگر قالب مربع لاتین باشد چون بلوک بندی ستونی هم اضافه میشود، پنج ستون تعریف خواهیم داشت.

a	b	c	rep	y
1	1	1	1	14705.66
1	1	1	2	14838.99
1	1	1	3	14066.46
1	1	2	1	17559.21
1	1	2	2	16500.35
1	1	2	3	15441.49
1	1	3	1	18416.48
1	1	3	2	17500.00
1	1	3	3	18600.00
1	2	1	1	15337.39
1	2	1	2	16621.67
1	2	1	3	14053.12
1	2	2	1	16053.00
1	2	2	2	15115.22
1	2	2	3	18961.00
1	2	3	1	18437.55
1	2	3	2	22949.81
1	2	3	3	19828.07
2	1	1	1	18396.79
2	1	1	2	16936.90
2	1	1	3	21290.28
2	1	2	1	19797.70
2	1	2	2	18943.21
2	1	2	3	22993.27
2	1	3	1	21308.00
2	1	3	2	22498.65
2	1	3	3	27618.00
2	2	1	1	20731.36
2	2	1	2	21037.00
2	2	1	3	18655.00
2	2	2	1	21441.56
2	2	2	2	24963.25
2	2	2	3	22257.00
2	2	3	1	26117.98
2	2	3	2	28201.38
2	2	3	3	29361.70

برنامه فاکتوریل سه عامله

بر پایه طرح کاملا تصادفی:

دقت شود این برنامه ها همانند دو فاکتوره ها هست و فقط عامل سوم به همراه اثرات متقابلش اضافه می شوند. در این آزمایشات چون تعداد اثرات متقابل بالا می رود از برنامه نویسی ساده تری در قسمت مدل استفاده شده که نیازی به نوشتن اثرات متقابل در مدل و مقایسات میانگین نباشد. این نوع برنامه نویسی برای دو فاکتوره ها قابل اجراست اما در سه فاکتوره به بالا کار را راحت می کند.

```
data;
input a b c rep y;
cards;

;
proc glm;
class a b c rep;
model y=a b c a|b|c/ss3;
lsmeans a|b|c/lines;
run; quit
```

بر پایه طرح بلوک های کامل تصادفی:

```
model y=rep a b c a|b|c/ss3;
```

بر پایه طرح مربع لاتین:

در این طرح علاوه بر مدل یک ستون تعریف هم اضافه می شود که علاوه بر مدل بالتبع بایستی در قسمتهای input و class هم لحاظ شود.

برنامه فاکتوریل-اسپیلیت پلات:

در قالب بلوک

```
data;
input a b c rep y;
cards;

;
proc glm;
class a b c rep;
model y=rep a b a*b rep(a b) c a*c b*c a*b*c/ss3;
test h=rep a b a*b e=rep(a b);
means a b/lsd e=rep(a b);
lsmeans a*b/lines e=rep(a b);
means c/lsd;
lsmeans a*c b*c a*b*c/lines;
run; quit;
```

جالب است بدانید در این طرح (فاکتوریل-اسپیلیت پلات) برای تغییر قالب طرح به حالت کاملا تصادفی فقط کافیست rep را از مدل حذف کنید و فرمت خطای اول تغییری نمیکند ولی SAS متوجه می شود قالب طرح عوض شده و خطا را کاملا تصادفی محاسبه می کند. پس بر خلاف آزمایشات قبلی نیازی نیست فرمول نویسی خطا تغییر کند فقط عامل rep کم و زیاد می شود.

برنامه اسپیلیت-فاکتوریل:

در قالب بلوک

```
data;
input a b c rep y;
cards;

;
proc glm;
class a b c rep;
model y=rep a rep*a b c a*b a*c b*c a*b*c/ss3;
test h=rep a e=rep*a;
means a/lsd e=rep*a;
means b c/lsd;
lsmeans a*b a*c b*c a*b*c/lines;
run; quit;
```

در قالب کاملا تصادفی مدل اینگونه می شود (هم خطا و هم نبود rep):

```
model y=a rep(a) b c a*b a*c b*c a*b*c/ss3;
```

برنامه اسپیلیت-اسپیلیت پلات:

در قالب بلوک

```
data;
input a b c rep y;
cards;

;
proc glm;
class a b c rep;
model y=rep a rep*a b a*b rep*b(a) c a*c b*c a*b*c/ss3;
test h=rep a e=rep*a;
test h=b a*b e=rep*b(a);
means a/lsd e=rep*a;
means b/lsd e=rep*b(a);
lsmeans a*b/lines e=rep*b(a);
means c/lsd;
lsmeans a*c b*c a*b*c/lines;
run; quit;
```

برنامه اسپیلیت-اسپیلیت پلات در زمان:

در قالب بلوک

این طرح نسبت به اسپیلیت-اسپیلیت پلات تنهای یک خطای اضافه تر دارد که عوامل c و a*c را آزمون می کند. زیرا که عامل c چیزی نیست جز زمان (در اسپیلیت پلات ها در زمان، عامل زمان فرعیترین عامل است در حالی که در تجزیه مرکب عامل اصلی خواهد بود).

```
data;
input a b c rep y;
cards;

;
proc glm;
class a b c rep;
model y=rep a rep*a b a*b rep*b(a) c a*c rep*c(a) b*c a*b*c/ss3;
test h=rep a e=rep*a;
test h=b a*b e=rep*b(a);
test h=c a*c e=rep*c(a);
means a/lsd e=rep*a;
means b/lsd e=rep*b(a);
means c/lsd e=rep*c(a);
lsmeans a*b/lines e=rep*b(a);
lsmeans a*c/lines e=rep*c(a);
lsmeans b*c a*b*c/lines;
run; quit;
```

موفق باشید